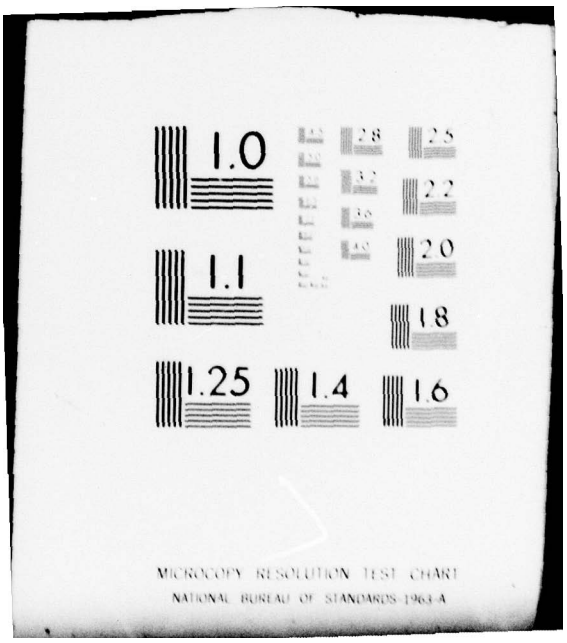
AD-A075 010 APPLIED ELECTRONIC RESEARCH CORP GAINESVILLE FL F/G 20/12
TWO-DIMENSIONAL AVALANCHE BREAKDOWN IN PT-SI SCHOTTKY BARRIER D--ETC(U)
JUL 79 D P KENNEDY F19628-77-C-0220
UNCLASSIFIED RADC-TR-79-189 NL

1 OF 2
AD A075010


1.0
1.1
1.25
1.4
1.6
2.8
2.5
2.2
2.0
1.8
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

RADC-TR-79-189
Final Technical Report
July 1979

LEVEL II

TWO-DIMENSIONAL AVALANCHE BREAKDOWN IN Pt-Si SCHOTTKY BARRIER DIODES

Applied Electronic Research Corporation

David P. Kennedy

ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441

This report has been reviewed by the RADC Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RADC-TR-79-189 has been reviewed and is approved for publication.

APPROVED: Sven A. Roosild

SVEN A. ROOSILD
Project Engineer

APPROVED: Robert M. Barrett

ROBERT M. BARRETT
Director, Solid State Sciences Division

FOR THE COMMANDER: John P. Huss

JOHN P. HUSS
Acting Chief, Plans Office

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (ESE), Hanscom AFB MA 01731. This will assist us in maintaining a current mailing list.

Do not return this copy. Retain or destroy.

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DDC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: RADC-TR-79-189	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): TWO-DIMENSIONAL AVALANCHE BREAKDOWN IN Pt-Si SCHOTTKY BARRIER DIODES,	5. TYPE OF REPORT & PERIOD COVERED: Final Technical Report
	6. PERFORMING ORG. REPORT NUMBER: N/A
7. AUTHOR(s): David P. Kennedy	8. CONTRACT OR GRANT NUMBER(s): F19628-77-C-0220
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Applied Electronic Research Corp. 2227 N. W. 16th Avenue Gainesville FL 32605	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 61102F 2305J125
11. CONTROLLING OFFICE NAME AND ADDRESS: Deputy for Electronic Technology (RADC/ESE) Hanscom AFB MA 01731	12. REPORT DATE: July 1979
	13. NUMBER OF PAGES: 146
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Same	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE: N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES

RADC Project Engineer: Sven A. Roosild (RADC/ESE)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Schottky barriers
Breakdown
Multiplication

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Under Contract No. F19628-77-C-0220 Applied Electronic Research Corporation undertook the development of a two-dimensional computer model for calculating avalanche breakdown in a Pt-Si Schottky barrier diode. A principal objective of this program was to develop such a computer program, and make the source-code and operational data available to other workers in the field. Further, as a means to verify the applicability of this program junction, breakdown computations were to be undertaken throughout a wide range of physical parameters and ambient temperatures.

DD FORM 1 JAN 73 1473

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

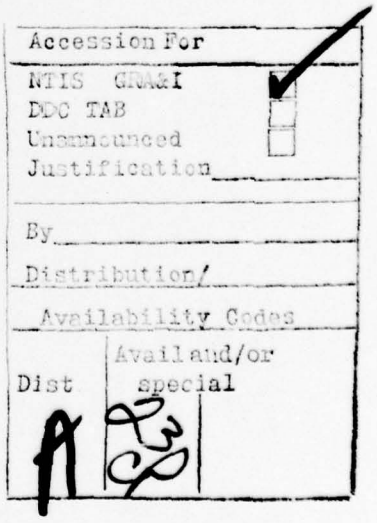
393479

Evaluation

The Final Report on Two-Dimensional Avalanche Breakdown in Pt-Si Schottky Barrier Diodes represents a half-man year effort over 12 months to calculate numerically the breakdown characteristics of Schottky barrier diodes on p-type silicon. The results clearly indicate that avalanche breakdown is not the appropriate mechanism to explain the experimentally observed reverse characteristics of these diodes; a far closer agreement is achieved with calculations based on a one carrier multiplication process. Preliminary considerations also indicate that it is inappropriate to apply Schottky barriers on n-type theoretical treatment to Schottky barrier on p-type diodes. In the former case tunneling may be an important leakage component. The findings of this contract have important applications to Schottky barrier based 3-5 micrometer infrared detector staring arrays as well as low temperature electronics.

Sven A. Roosild

SVEN A. ROOSILD
Project Engineer



Summary and Major Results

A two-dimensional computer model has been developed for avalanche breakdown in a Pt-Si Schottky barrier diode, assuming either p-type or n-type semiconductor material. After completing this model, comparisons between experiment and theory suggested a need to undertake model modifications; assuming carrier multiplication processes other than a classical Townsend avalanche processes. Those modifications were undertaken and completed. As a consequence, two different computer models are being made available: first, a model assuming classical Townsend avalanche breakdown and, second, a model assuming one-carrier multiplication within a pt-Si space-charge layer.

Both of these computer programs utilize finite-difference methods within a two-dimensional grid containing non-equal spacing. Thereby, substantial computational accuracy is obtained throughout regions of the structure containing large values of electric field and a significant level of carrier multiplication. User options permit adequate variation of the geometrical and physical parameters in this model to make it valuable tool for fundamental studies associated with two-dimensional Schottky barrier operation.

Computational techniques have been employed that appear to reduce overall computation time to a satisfactory level. Specifically, these computer programs are designed to undertake a search for the carrier trajectory producing a maximum of carrier multiplication, at a given applied voltage; in contrast with undertaking a calculation of all carrier trajectories and their respective levels of carrier multiplication. Further, the system of finite difference equations generated during a given two-dimensional solution of Poisson's equation are solved using efficient space-matrix techniques. Stone's method is applied in an iterative loop for calculating the potential distribution throughout the space-charge layer of a Pt-Si Schottky barrier diode.

Avalanche breakdown calculations were undertaken for physical parameters typical of devices being fabricated in laboratory experiments. Important differences were observed between experiment and theory: the

calculated breakdown voltage was higher than observed in the laboratory, and the calculated avalanche breakdown process was substantially more abrupt. It was these results that suggested a change in the model for carrier multiplication--to a one-carrier process. This model modification did, indeed, produce a soft breakdown characteristic and better agreement with experiment.

A one-carrier multiplication process yields a reverse current that increases nearly exponentially with voltage. It was found that some laboratory devices do indeed exhibit an exponential current/voltage relation throughout a substantial range of reverse current. Unfortunately, little inference can be drawn from this observation; theory also predicts that carrier tunneling at the metal/Si interface would exhibit a similar current-voltage relation. For this reason it is suggested that the reverse current observed in a p-type Pt-Si diode does not exhibit those characteristics expected from an avalanche breakdown process. Instead, it is suggested that this reverse current could arise from either one-carrier multiplication within the space-charge layer or from carrier tunneling at the metal-Si interface.

TABLE OF CONTENTS

Table of Contents (Continued)

LIST OF ILLUSTRATIONS

LIST OF TABLES

CHAPTER I

A Two-Dimensional Computer Model for Avalanche Breakdown in Schottky Barrier Diodes

1.0 Introduction

Schottky barriers formed with metal-silicide compounds in contact with silicon are known, under some conditions, to exhibit "soft" reverse breakdown characteristics. That is to say the reverse current is observed to increase gradually, rather than abruptly, as the barrier is biased in the reverse direction. This phenomenon is observed, for example, with platinum-silicide Schottky barriers, using p-type silicon and operated at liquid nitrogen temperature (77° K).

Possible causes for this soft breakdown phenomenon are: First, unlike barriers formed by the juncture of a pure metal (such as gold) in contact with silicon, the kinetics of the growth of a metal-silicide compound produces a metal-silicide region which penetrates into the silicon substrate in a manner similar to a shallow diffusion [1]. Thus, two-dimensional mechanisms exist, wherein high electric fields may develop at the periphery of the barrier causing breakdown to occur, due to avalanche multiplication. Second, mechanical stress at the interface [2, 3] can enhance this breakdown phenomenon. Third, the existence of surface states could lead to enhanced breakdown. Fourth, quantum mechanical tunneling of carriers through the potential barrier formed at the metal-silicide-silicon interface also represents possible cause of soft breakdown.

Work performed under this program addressed the following tasks:

A.--Develop a two-dimensional computer solution for Poisson's equation, using boundary conditions that include a metal-silicide Schottky barrier on p-type silicon. Further, the surface of this structure is bounded by a thermal oxide containing positive charges, accounting for oxide-silicon interface states.

B.--Develop a two-dimensional computer solution for the Townsend carrier multiplication process.

C.--Calculate the breakdown voltage of a Schottky barrier on p-type silicon as a function of the following parameters:

1. Barrier height (from 0.25 ev to 0.35 ev)
2. Resistivity (from 1.0 to 100 chm cm)
3. Ox/Si interface charge (10^{-7} to 10^{-9} coul:/cm^2)
4. Temperature ($77^{o}K$ to $300^{o}K$)

D.--Extend the above model to permit the treatment of N-type silicon structures of the same type described in A, above.

2.0 Description of the Computer Model

The computer model treats either p-type or N-type substrate materials, the distinction being determined through a "flag" incorporated in the input data parameters in a manner to be clarified in a later subsection. It is capable of treating arbitrary doping densities and temperatures, also specified as input parameters. Considerable flexibility is also incorporated insofar as the geometry is concerned.

2.01 Basic Geometry

Figure 1 illustrates the geometry assumed in this computer model. A rectangular region is defined, containing a metal-silicide region of constant potential (i.e. it is treated as if it were a metal), having the value (V_{bi} + V). Here V_{bi} is the "built-in" equilibrium potential between the two regions, while V is the externally applied reverse bias voltage. This metal-silicide region is considered to be a rectangle, with dimensions X_m by Y_m, with a circular lower right corner. The overall rectangular region extends to the right of and below the metal-silicide region by 1.25 depletion widths; thereby, we are assured that all boundaries are located in charge-neutral substrate material. Our definition of a "depletion width" hinges upon the geometric parameters X_m, Y_m, and r_m. In the event that all three are equal, and if the silicon-oxide surface charge Q_{ss} is zero, then a one-dimensional depletion solution of Poisson's equation (in cylindrical coordinates) is used to establish the required width. If, instead, any of these conditions are violated (thereby destroying the circular symmetry) planar depletion theory is applied to determine the depletion layer width.

The left border (extending below the metal-silicide region) is treated as a symmetry plane along which the normal derivative of the electrostatic potential is set to zero. A similar boundary is used for the upper border (extending to the right of Pt-Si region), except that the normal derivative

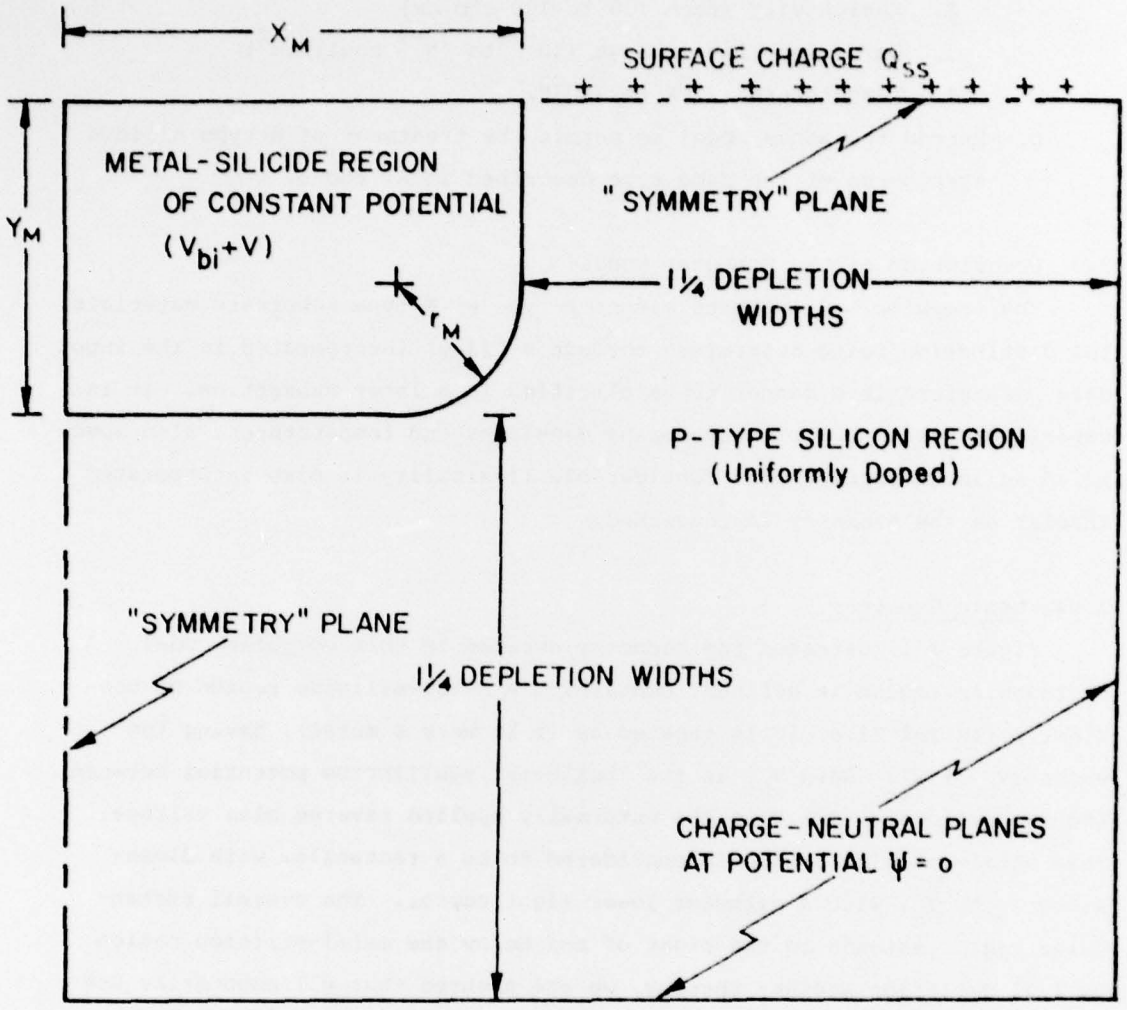


Fig. 1
Geometrical Model used in this Computer Program

of electrostatic potential becomes discontinuous when the surface charge Q_{ss} is nonzero.

This computer model deals with a graded lattice of mesh points overlayed upon the foregoing rectangular region. Lattice grading is automatically implemented by a part of this computer program. Such grading has been designed to make the density of mesh points highest in the vicinity of the metal-silicide-silicon interface; thereby enhancing the accuracy of the finite-differences algorithms throughout the region where avalanche multiplication occurs. Figure 2 illustrates such a graded lattice for the case where $X_m = Y_m$, using an array of 31 by 31 mesh points.

2.02 Normalized Variables

With a few exceptions which are clearly flagged by COMMENT statements, this computer model deals with normalized variables. These variables are defined in terms of the following normalization parameters:

1) Distances are normalized with respect to the depletion width, $W_D = \sqrt{2E(V_{bi} + V) / (qN_a)}$, obtained from planar depletion theory. Here N_a is the ionized acceptor density at temperature T. (This is replaced by the ionized donor density N_d, when modeling N-type structures.)
2) The electrostatic potential is normalized with respect to the voltage $(V_{bi} + V)$.
3) All charge densities are normalized with respect to the ionized acceptor density N_a or donor density N_d, as the case may be.

2.03 Determination of the Electrostatic Potential ψ, and the corresponding Electric Field Components E_x & Ey.

Expressed in terms of the normalized variables, Poisson's equation becomes:

$$\nabla^2 \psi = 2[1 + n - p] = Q \qquad (1.01)$$

where ψ is the electrostatic potential, while p and n are the normalized hole and electron densities, respectively. The former, in dimensionless variables, is related to ψ through the Boltzmann relationship:

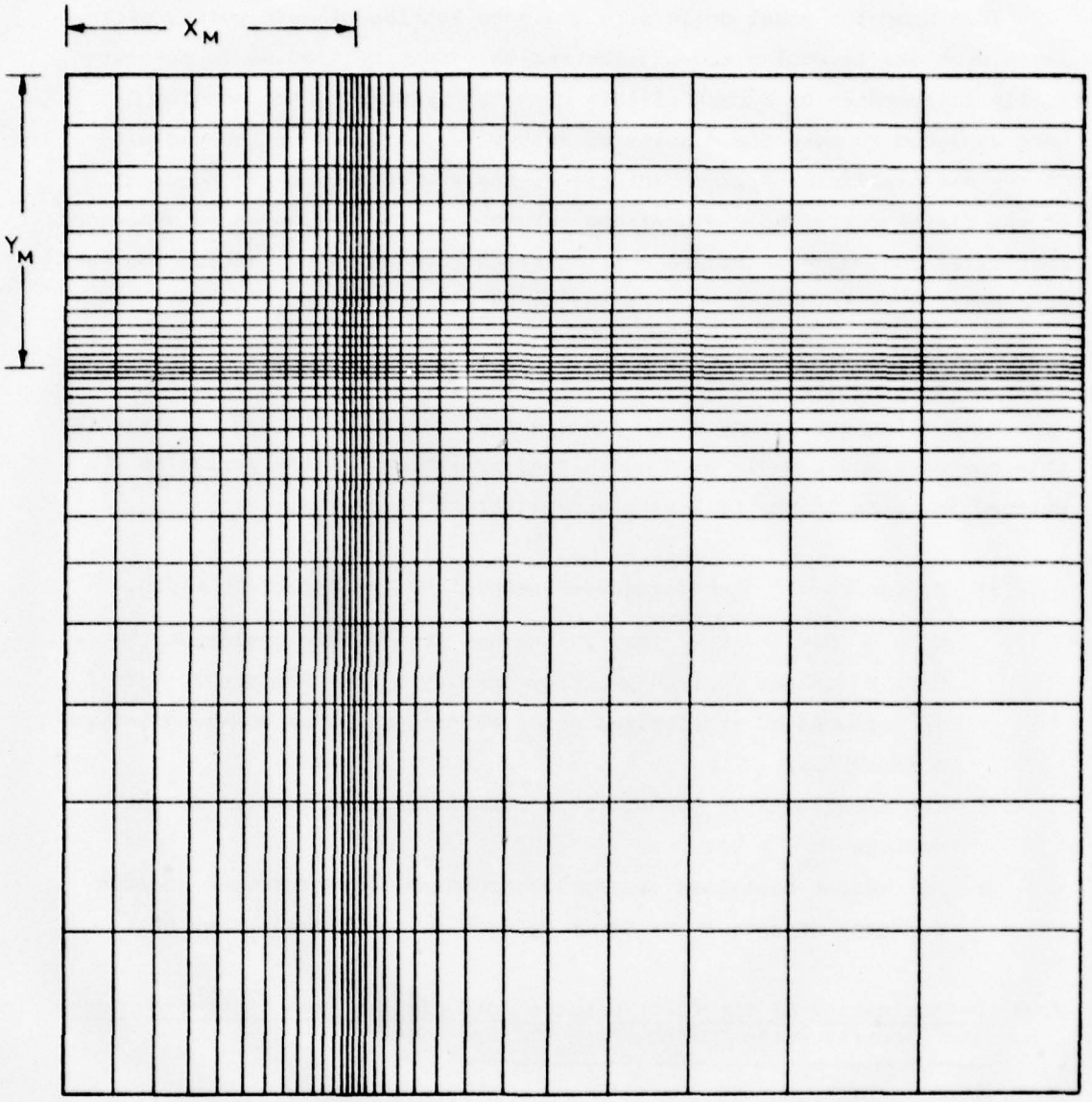


Fig. 2
Illustration of Lattice Grading Established in Subroutine GRID

$$p = \exp(-q[V_{bi} + V]\,\psi/kT), \qquad (1.02)$$

which implies nondegenerate doping of the substrate material. The latter is similarly expressed by:

$$n = n_i^2 \exp(q[V_{bi} + V)\,\psi - V]/kT), \qquad (1.03)$$

where n_i is the normalized intrinsic density at temperature T.

Equations (1.02) and (1.03) are based upon an approximation which ignores gradients in the hole and electron quasi-Fermi potentials; both are assumed to be independent of X and Y throughout the silicon substrate region. This assumption, widely used in p-n junction breakdown calculations, is justified by the success of such calculations in predicting breakdown voltages in excellent agreement with experimental results. When the reverse bias voltage V is nonzero, (1.03) implies that the electron quasi-Fermi potential differs from that for holes by the value V. Note that (1.03) correctly gives $n = n_i^2 \exp(qV_{bi}/kT)$ at the metal-silicide-silicon interface, where ψ has the value of unity in normalized units. Further, (1.02) and (1.03) correctly yield the mass action law, $pn = n_i^2$, under equilibrium conditions (i.e. when $V = 0$).

The nonlinear problem posed by (1.01) through (1.03) is solved iteratively by the computer model. An initial "guess" is first obtained for ψ, using depletion theory. Equations (1.02) and (1.03) then yield initial estimates for p and n at each lattice mesh point, and therefore for the right hand side of (1.01). That equation, when expressed in finite-difference form for a lattice where dimensions are m by n, transforms into m times n coupled linear equations for ψ at each mesh point. Expressed in matrix form, the resultant matrix is very sparse, having only five nonzero diagonals. Stone's iterative matrix factorization method [4] is then utilized to obtain a revised estimate for ψ at each of the mesh points. Then (1.02) and (1.03) provide corresponding revised estimates for p and n. This procedure is applied repetitively until self-consistent solutions of (2.01) through (2.03) result. The implementation of this iterative process utilizes Grummel's algorithm [5], as modified by Sutherland [6], to enhance its convergence.

With a self-consistent solution of Poisson's equation in hand, the normalized electric field components E_x and E_y are obtained using finite

differences. These electric field components are stored two arrays, for subsequent use in calculating the free carrier trajectories, and the Townsend multiplication integrals.

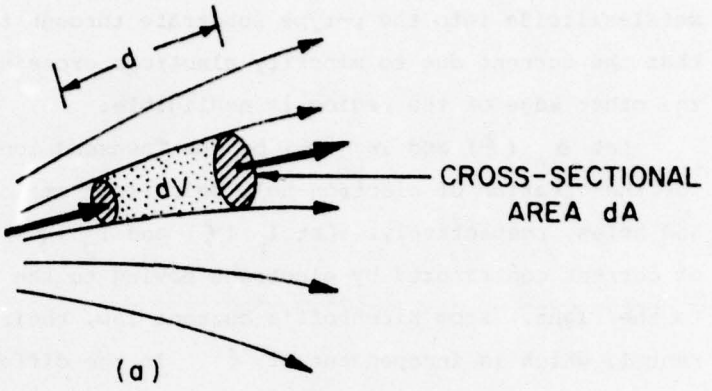
2.04 Determination of the Two-Dimensional Free Carrier Trajectories

The evaluation of the Townsend multiplication integrals requires knowledge of the curvilinear trajectories traced out by the charge carriers involved in that process. Since significant multiplication occurs only in regions where the electric field (unnormalized) exceeds 2×10^5 volts/cm, the effects of diffusion forces are negligible in comparison with the electrostatic forces acting upon said carriers. Thus, the resultant velocity vector $\bar{v}$ is colinear with the electric field vector $\bar{E}$ ($\bar{v} = \mu \bar{E}$ in normalized units, with μ the normalized carrier mobility.) As a result, the paths of the carriers coincide everywhere with electrostatic flux lines, regardless of the numerical value of their mobility. Therefore, in determining those paths, it is sufficient to set the normalized mobility equal to unity, and this is what is done in the computer model.

The two-dimensional equations of motion are solved using a subprogram which implements the Adams-Bashforth-Moulton predictor-corrector algorithm [1] with time of flight as the independent variable. This subprogram provides for automatic interval halving (and doubling), controlled by monitoring the difference between the predicted and corrected values of the charge carrier's position as a function of the time of flight.

2.05 Determination of the Townsend Multiplication Ratio M

Figure 3 depicts a region in which two-dimensional Townsend multiplication processes are in progress. Part (a) of that figure sketches a few curvilinear streamlines, and singles out a small differential volume dV, where "sidewalls" coincide with the streamlines, and the ends are perpendicular to the streamlines. Although such streamlines for two-dimensional flow may converge or diverge, conservation of electric current requires that the net current entering one end of such a differential volume must equal the net current leaving it through the other end. From such an argument it is concluded that the Townsend multiplication process can be treated as a one-dimensional problem in the spacial coordinate ξ, which is singly curvilinear distance measured along such a streamline. Part (b) of the figure depicts this one-dimensional problem in ξ.



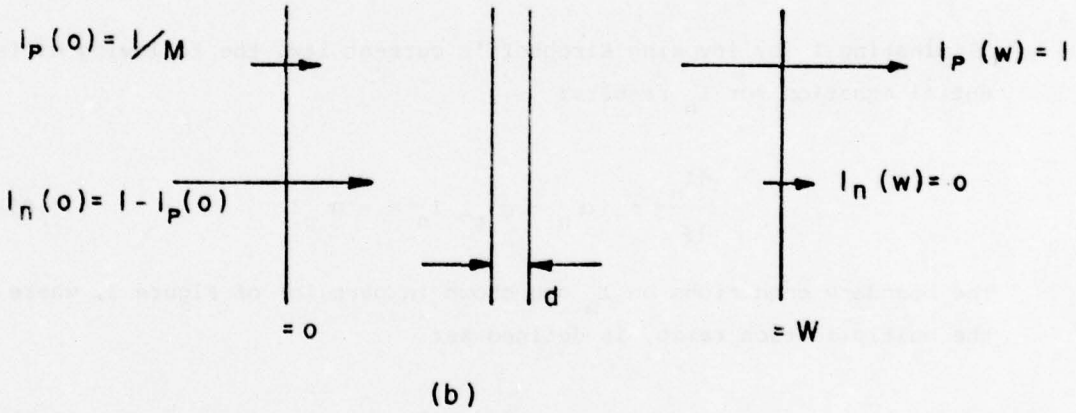


Fig. 3
Stream Lines in A Semiconductor Material
a) Two Spatial Dimensions
b) One-Dimension Approximation

Next we discuss multiplication in terms of p-type material, with the understanding that an interchange of subscripts p and n will make the results applicable to n-type material as well. We assume that the multiplication process is initiated by the injection of majority holes from the metal-silicide into the p-type substrate through the plane $\xi = 0$, and that the current due to minority electrons crossing the plane $\xi = W$ at the other edge of the region is negligible.

Let $\alpha_n(\xi)$ and $\alpha_p(\xi)$ be the Townsend ionization coefficients for the creation of electron-hole pairs (per cm) of travel of electrons and holes, respectively. Let $I_n(\xi)$ and $I_p(\xi)$ be the local components of current contributed by electrons moving to the left and holes moving to the right. From Kirchhoff's current law, their sum is the total current I, which is independent of ξ. In the differential distance $d\xi$, both I_n and I_p change, due the creation of additional electron-hole pairs:

$$dI_p(\xi) = dI_n(\xi) = [\alpha_n(\xi)\, I_n(\xi) + \alpha_p(\xi)\, I_p(\xi)]d\xi \qquad (1.04)$$

Eliminating I_p by invoking Kirchhoff's current law, the following differential equation for I_n results:

$$\frac{dI_n}{d\xi} + (\alpha_n - \alpha_p)\, I_n = -\alpha_p I \qquad (1.05)$$

The boundary conditions on I_n are shown in part (b) of Figure 3, where M, the multiplication ratio, is defined as:

$$M = I/I_p(0) \qquad (1.06)$$

The solution of (1.05) which satisfies those boundary conditions leads to the following relationship for M:

$$M = \left\{ \frac{1}{1 - \int_0^W \alpha_p(\xi) \exp\left\{ \int_0^\xi [\alpha_n(\xi') - \alpha_p(\xi')]d\xi' \right\} d\xi} \right\} \qquad (1.07)$$

Inspection of this result reveals that M approaches an infinite value as the integral expression in the denominator approaches unity. Since the ξ-dependence of α_n and α_p in the integrand depends upon the magnitude and ξ-dependence of the electric field E which, in turn, depends upon the applied reverse bias voltage V, there will always exist a value for the latter which will cause M to approach infinite values. This is the condition known as avalanche breakdown, and the value of V which causes it is the reverse breakdown voltage. As will be seen in the next chapter, where the results obtained with the computer model are discussed, this breakdown phenomenon takes on a "soft" appearance in the reverse I-V characteristic curve only when the radius of curvature r_m of the metal-silicide region is very small.

The derivation leading to (1.07) for M assumed a two-carrier multiplication process wherein both electrons and holes contribute to the avalanche mechanism. For silicon, however, experimental data for the ionization coefficients α_n and α_p as a function of the electric field E show that α_p is more than an order of magnitude less than α_n, except for very large electric fields [8]. Thus, electrons, once created by impact ionization, are much more effective than holes in creating additional electron-hole pairs. This leads one to question whether or not a "soft" breakdown characteristic might result due to single carrier multiplication, wherein only electrons are responsible for the creation of additional electron-hole pairs. In other words, what would result if $\alpha_p(\xi)$ were assumed to be zero?

Inspection of (1.07) shows that merely setting $\alpha_p(\xi)$ to zero will not work in this particular equation; no multiplication is obtained (M = 1). This conclusion arises from the fact that when $\alpha_p(\xi) = 0$ (in the form of $I_p(0)$ in the boundary conditions, with $I_n(w) = 0$) only holes can initiate the avalanche process. Setting $\alpha_p(\xi) = 0$ and retaining $I_n(w) = 0$, implies than an electron-hole pair can never be created, and since only electrons so created are assumed to contribute to the creation of additional electron-hole pairs, unity values of M are the natural consequence. The assumption of negligible electron current at $\xi = W$ must be removed if there is to be any resultant multiplication when $\alpha_p(\xi) = 0$.

To explore the implications of single carrier multiplication due to electrons alone, insofar as "soft" breakdown is concerned, we elected to

treat the extreme case in which only electrons initiate the avalanche process, and cause the creation of additional electron-hole pairs. This is achieved by redefining M to be:

$$M = I/I_n\ (w) \tag{1.08}$$

$$I_p\ (0) = 0$$

$$I_n\ (0) = I \tag{1.09}$$

$$I_p\ (w) = I - I_n\ (w).$$

Then $\alpha_p\ (\xi)$ is set to zero in (1.05). The resulting expression for M becomes:

$$M = \exp\left\{\int_0^w \alpha_n(\xi)\,d\xi\right\} \tag{1.10}$$

This result clearly exhibits a "soft" breakdown characteristic inasmuch as M approaches infinite values only as the integral in (1.10) does so also. There is no finite reverse bias voltage V leading to an infinite current. Such a conclusion is not surprising; it was precisely such a conclusion that encouraged Townsend to postulate two-carrier multiplication.

There is a problem with this result, and this problem is associated with the redefinition of M (compare (1.06) with (1.09)). In a hole initiated multiplication process the reverse saturation current of the Schottky barrier is clearly a consequence of space-charge generated holes that move to the p-type substrate material. In the latter case, it is the current due to minority electrons flowing from the p-substrate to the metal-silicide. The former is many orders of magnitude larger than the latter, especially at reduced temperatures. At $77^{\circ}K$, for example, the intrinsic electron density in silicon is $5.22 \times 10^{-20}\ cm^{-3}$. With an acceptor density of $10^{15}\ cm^{-3}$, the corresponding equilibrium minority electron density in the charge-neutral substrate is calculated to be $2.73 \times 10^{-54}/cm^3$. Thus, in Fig. 3, the minority electron current crossing the plane $\xi = W$ should be a negligible factor in contributing to the reverse I-V characteristic.

From the foregoing discussion it is evident that single carrier multiplication, due to electrons, is an implausible mechanism in an ideal semiconductor structure. If, instead, substantial lattice damage exists (micro-cracks, strain, etc.), electron/hole generation could take place, and such damage could represent a significant source for electrons. For this reason (1.10) has been also implemented in the present computer model, and this expression leads to results differing from (1.07).

The computer model functions as follows when evaluating the above expressions for M. A search is first made to ascertain the location of the maximum electric field along the metal-silicide-silicon interface. Three trajectories are then determined, the middle one initiating at the point of maximum E, the other two bracketing it. Equation (1.07) is next evaluated along all three paths, and a test is made to ascertain which of the three produces maximum M. If not the middle path, the pattern of three paths is shifted left or right along the interface, accordingly, and the process is repeated until the middle path produces a maximum M for a given applied voltage V. In the special case that the geometry under study has circular symmetry (i.e. $X_m = Y_m = r_m$, and $Q_{ss} = 0$), this trial and error search procedure is bypassed. (The symmetry then assures that all paths will produce the same M, and only a single trajectory need be treated. The one selected is that which initiates in the diagonal plane of symmetry.)

Two versions of the subprogram for evaluating the breakdown phenomenon have been developed. The first subprogram, which is designed to find a voltage V producing an infinite value of M, requires a substantially greater computation time. The second subprogram is designed to provide data for plotting M versus V.

In the first version, an assumed applied voltage V is varied, using an iterative process, until the two-carrier multiplication factor M changes sign. This change of sign signifies that the breakdown voltage has been bracketed. During iteration the two-dimensional potential distribution, and the electric field distribution, are scaled with the revised estimates for V, in accordance with their dependency upon V from planar depletion theory. This scaling is done to avoid repeated new solutions of Poisson's equation, and repeated new trajectory calculations. Once having "zeroed" in on a correct V, however, the entire procedure is then repeated iteratively. New self-consistent solutions for the potential and electric

field are generated, and new trajectory calculations are undertaken to find a path of maximum M and a revised estimate for V. This iterative process is continued until the most recent estimate of the breakdown voltage is within a specified percentage of its previous value.

With the second version of this computer program, the applied voltage V is varied through a given range, in given increments, and the corresponding values of M (using both (1.07) and (1.10) are determined and tabulated. This version is much more useful for exploring "soft" breakdown tendencies, supplying as it does direct information for plotting curves showing M versus V.

A simplified version of (1.07) is used for avalanche breakdown calculations. In this version (1.07) α_n and α_p are assumed equal, having a dependence upon electric field as determined by Kennedy and O'Brien [9], and found to produce results in good agreement with experiment for a wide variety of silicon p-n junction devices. The Kennedy-O'Brien value of α, for room temperature, is corrected for the specified operating temperature T fitting to the theoretical results of Baraff [10], as discussed by Crowell and Sze [11]. The second version of this computer program, discussed above, utilizes Crowell and Sze's formulas directly to obtain estimates for both α_n and α_p for use in (1.07) and (1.10). As a consequence of these differences, the two approaches yield slightly different estimates for the voltage leading to avalanche breakdown.

2.06 Calculation of the Temperature-Dependent Parameters

Figure 4 illustrates the energy band diagram (in electron volts) for a Schottky barrier on p-type silicon, and defines the intrinsic and extrinsic Fermi levels E_i and E_f, the valence and conduction band edge energies E_v and E_c, the energy gap E_g, the acceptor impurity level E_a, (all referred to the charge-neutral substrate region), as well as the Schottky barrier height ϕ_B and the "built-in" potential difference V_{bi}. The interrelationship between these fundamental parameters, and the dependence of the intrinsic and extrinsic majority hole densities upon them, are sensitive to temperature (in some instances dramatically) as well as to the doping density. These dependencies are taken into account in a subprogram of the computer model in the manner next to be described.

a) Energy Gap.--The energy gap E_g is slightly dependent upon temperature. The following formula is an empirical fit to Figure 10.2 of [12]:

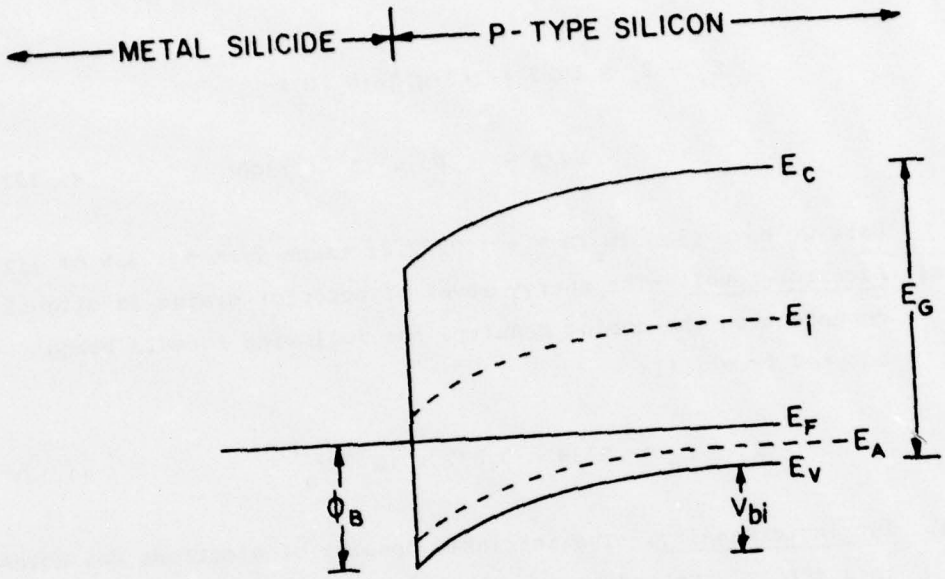


Fig. 4
Energy Band Diagram
For A Schottky Barrier on p-type Silicon

$$E_g = 1.165 - 7.242 \times 10^{-3}(T/300) - 3.664 \times 10^{-2}(T/300)^2 \quad (1.11)$$

b) Intrinsic Fermi Level.--The intrinsic Fermi level E_i is essentially located at the midgap position, deviating from same in a manner slightly dependent upon temperature, as well as upon the ratio of the effective density of states at the conduction and valence band edges:

$$E_i - E_v = Eg/2 + (kT/q)\ln(N_v/N_c)$$

$$= Eg/2 - 1.306 \times 10^{-2}(T/300) \quad (1.12)$$

Here we have used $(N_v/N_c) = 1.02/2.8$, taken from pg. 359 of [12].

c) Acceptor Level.--The energy level of acceptor states is slightly dependent on the doping density, the following formula being adopted from [13]:

$$E_a - E_v = .0438 - 3.037 \times 10^{-8} N_a^{1/3} \quad (1.13)$$

d) Intrinsic Density.--The intrinsic density of electrons and holes is a strongly dependent function of temperature [14]:

$$n_i = 3.925 \times 10^{19} \left(\frac{T}{300}\right)^{3/2} \exp\left\{-qE_g/2kT\right\} . \quad (1.14)$$

Here the leading numerical constant has been chosen so as to cause n_i to equal 1.5×10^{10} cm^{-3} when $T = 300^{\circ}K$.

e) Bulk Hole Density.--The hole density within the charge-neutral substrate region is related to the intrinsic density and to the Fermi energies E_i and E_f by the well-known relationship:

$$p = n_i \exp\left\{q(E_i - E_f)kT\right\} \quad (1.15)$$

This, too, is strongly dependent upon temperature.

f) Ionized Acceptor Density.--The ionized acceptor density is moderately dependent upon temperature, the following being adopted from [13]:

$$N_a = \left[\frac{N_a}{1 + \left\{4 + 2\exp\left(\frac{-q\Delta}{kT}\right)\right\} \exp\left\{q\,(E_a - E_f)/kT\right\}} \right] \qquad (1.16)$$

where Δ = .044 eV, and accounts for the spin orbit splitting at the valence band maximum in silicon.

The crux of the temperature dependence of the equilibrium parameters is seated in (1.14) through (1.16), with the latter two of these equations involving the extrinsic Fermi level E_f, which is unknown a-priori. That level must self-adjust to satisfy the requirement of charge neutrality in the bulk substrate region. Given the operating temperature T, self-consistent solutions of (1.14) through (1.16) are required which satisfy the condition:

$$N_a + p = 0 \qquad (1.17)$$

(Here, we ignore the small additional charge due to the presence of minority electrons, which constitutes a negligible correction.)

The subprogram of the computer model which treats the determination of the above equilibrium parameters, given the doping density N_a and the operating temperature T as input data, determines this self-consistent solution of (1.14) through (1.16), using iterative methods.

TABLE I illustrates the temperature dependence of the intrinsic density n_i, the ionized acceptor density N_a, and the "built-in" potential difference V_{bi}, for the case where the acceptor density N_a is 10^{15}, and the Schottky barrier height ϕ_B is 0.27 eV (typical of a platinum-silicide on p-type silicon structure).

$T(^{o}K)$	$n_i(cm^{-3})$	$N_a(cm^{-3})$	V_{bi}(Volts)
50	6.73×10^{-41}	1.62×10^{14}	2.30×10^{-1}
100	4.61×10^{-11}	9.16×10^{14}	1.97×10^{-1}
150	6.08×10^{-1}	9.89×10^{14}	1.53×10^{-1}
200	8.08×10^{4}	9.97×10^{14}	1.07×10^{-1}
250	1.08×10^{8}	9.98×10^{14}	5.89×10^{-2}
300	1.50×10^{10}	9.99×10^{14}	9.60×10^{-3}
350	5.32×10^{11}	9.99×10^{14}	-4.08×10^{-2}

TABLE I -- Temperature Dependence of the Intrinsic Density n_i, the Ionized Acceptor Density N_a, and the Built-in Potential V_{bi} for a Schottky Barrier on p-type Silicon, for the Case $\phi_B = 0.27$ ev, $N_A = 10^{15}$ cm^{-3}

Examining this table, one notes the negligible density of minority carriers ($n \approx n_i^2/N_a$) in comparison with majority carriers ($p \approx N_a$) over the entire temperature range. One also notes the rapid decrease of the "built-in" potential V_{bi} as the temperature increases. The barrier becomes essentially "ohmic" at temperatures well below $300^{o}K$. One notes, also, the gradual "freeze-out" of ionized acceptors as the temperature decreases.

The discussion above has been directed toward p-type Schottky barriers. Minor modifications make it applicable to n-type structures as well:

a) One substitutes the n-type barrier height in place of the ϕ_B for p-type material. (The sum $\phi_{Bp} + \phi_{Bn}$ equals the energy gap E_g). One also substitutes N_d in place of N_a, and interchanges subscripts p and n.

b) One substitutes the ionized donor density N_d in lieu of the ionized acceptor density N_a. The temperature dependence of the former differs from that of the latter (eq. (1.16)), becoming:

$$N_d = \left\{ \frac{N_d}{1 + 2 \exp \left\{ q(E_f - E_d)/kT \right\}} \right\} \qquad (1.18)$$

Equation (1.18) differs from (1.16) on two counts: 1) There is no spin-orbit splitting in the conduction band; 2) The ground state degeneracy factor for the conduction band is 2, instead of 4.

CHAPTER II

Calculations of the Reverse Voltage Breakdown Characteristics of Metal-Silicide Schottky Barriers on P-Type Silicon

In this chapter we present results obtained with the computer model for p-type structures, assuming r_m, ϕ_B, N_a, Q_{ss}, and T are the given parameters.

1.0 The Effect of Barrier Radius of Curvature

Figure 5 illustrates the effect of barrier radius of curvature (r_m in Fig. 1) upon the multiplication ratio M. This figure shows M versus V, using both (1.07) and (1.10) for two-carrier and single carrier multiplication, respectively. These calculations were based upon an assumed N_a of 1.6×10^{15} (10 ohm-cm material), and ϕ_B = 0.27 ev (simulating a platinum-silicide barrier), at T = 77^{O}K. Circular symmetry was used in this calculation ($x_m = y_m = r_m$).

Figure 5 shows that breakdown occurs at progressively lower voltages, with a reduction in radius of curvature. Further, this sequence of calculations show that single carrier multiplication is "softer" in appearance than a two-carrier multiplication process, where $M = \infty$ at finite reverse voltages.

Figure 5 reveals an interesting trend, however. The smaller the radius of curvature, the more nearly do the curves for two-carrier and single-carrier multiplication coincide at low voltages and currents. Specifically, a two-carrier multiplication process becomes "soft" with a reduction in radius of curvature. The foregoing observation is further illustrated in Figure 6, which is a "blown-up" view of the $r_m = .01\ \mu$m, curve from Fig. 5.

From Fig. 6, it is suggested that a very shallow platinum-silicide layer could produce a very small radius of curvature and, hence, soft breakdown characteristics at small values of applied biasing voltage.

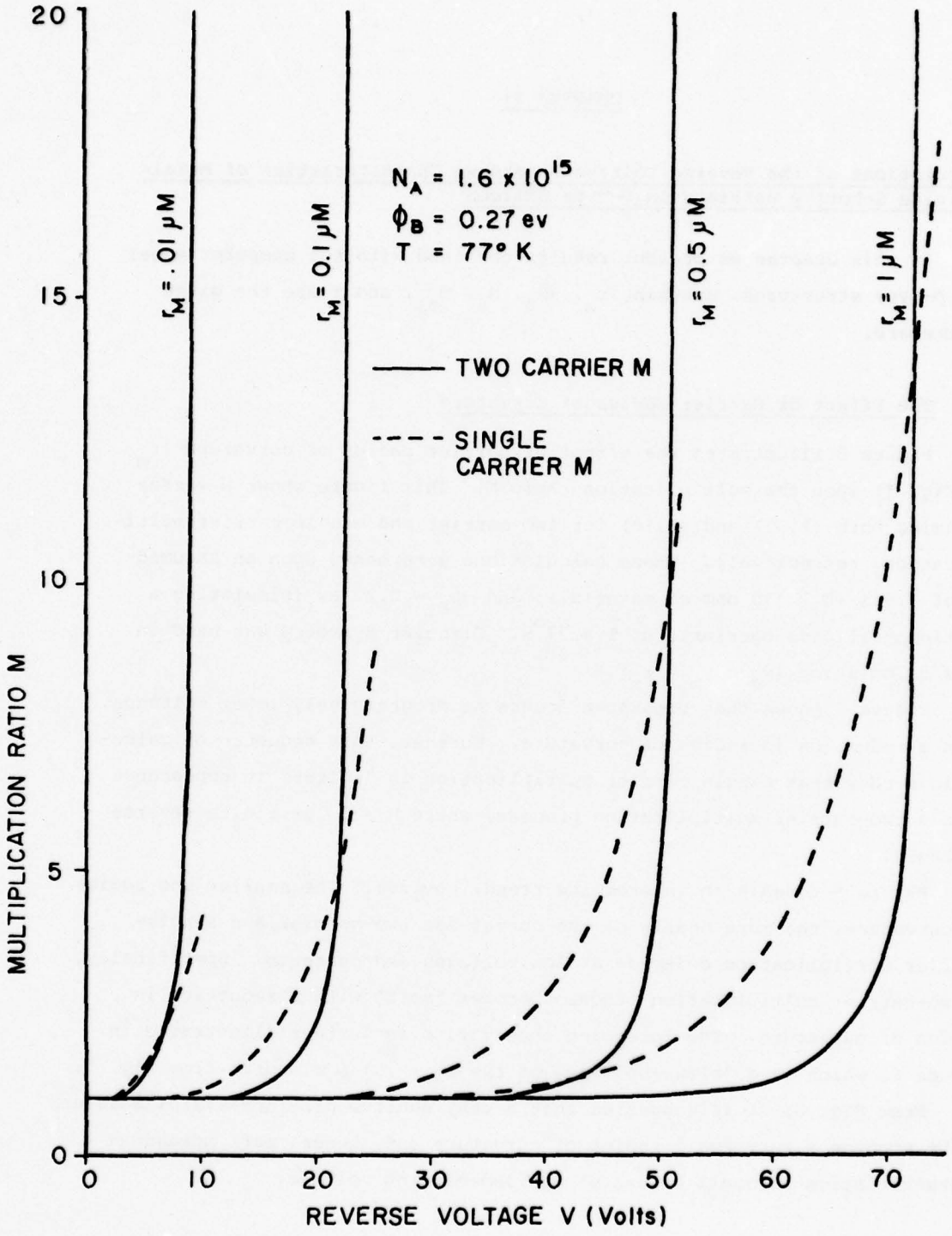


Fig. 5
Multiplication Ratio vs. Reverse Applied Voltage
for a Range of Junction Radius of Curvature

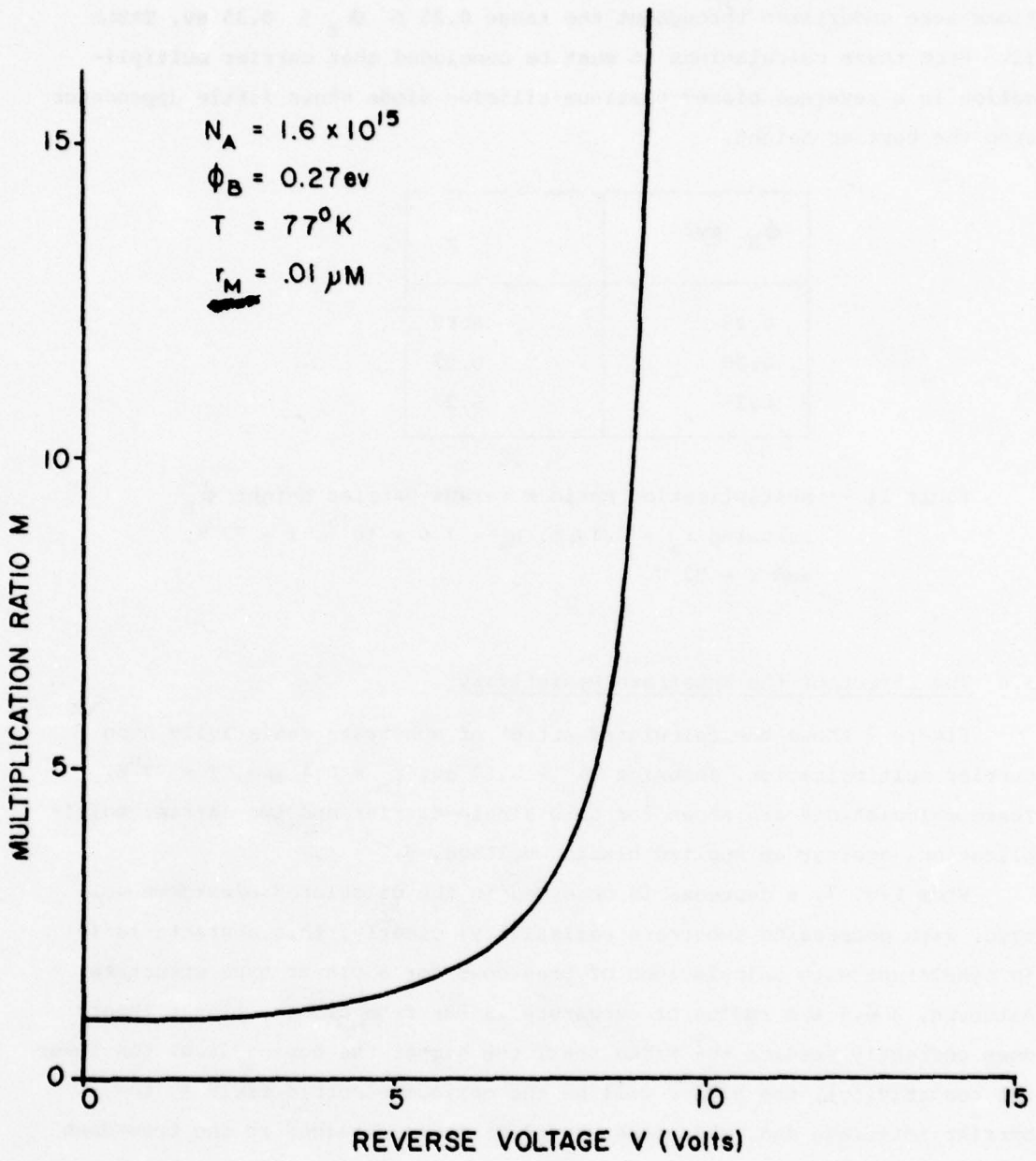


Fig. 6
Multiplication Ratio M vs. Applied Reverse Voltage,
Assuming the Radius of Curvature is 100 Å

2.0 The Effect of the Barrier Height

Breakdown calculations were undertaken to ascertain the sensitivity of carrier multiplication, M, on the barrier height ϕ_B. These calculations were undertaken throughout the range $0.25 \leq \phi_B \leq 0.35$ ev, TABLE II. From these calculations it must be concluded that carrier multiplication in a reversed biased platinum-silicide diode shows little dependence upon the barrier height.

ϕ_B (ev)	M
0.25	5.80
0.30	6.07
0.35	6.37

Table II -- Multiplication Ratio M versus Barrier Height ϕ_B assuming $r_m = 0.1 \mu m$, $N_a = 1.6 \times 10^{15}$, $T = 77^{\circ}K$, and V = 22 V.

3.0 The Effect of the Substrate Resistivity

Figure 7 shows the calculated effect of substrate resistivity upon carrier multiplication, assuming $\phi_B = 0.27$ ev, $r_m = 0.1$ μm, $T = 77^{\circ}K$. These calculations are shown for both single-carrier and two-carrier multiplication, against an applied biasing voltage, V.

From Fig. 7, a decrease is observed in the calculated breakdown voltage, with decreasing substrate resistivity; clearly, this characteristic is consistent with calculations of breakdown for a planar type structure. Although, a 0.1 μm radius of curvature is far from planar, planar theory does correctly predict the trend that, the higher the doping (i.e. the lower the resistivity), the higher will be the maximum electric field at the barrier interface and, therefore, a reduction is obtained in the breakdown voltage.

4.0 The Effect of Surface Charge at the Silicon-Silicon Dioxide Interface

The effective surface charge Q_{ss} is known to arise when SiO_2 is grown on the surface of silicon; this charge is known to be positive, and with

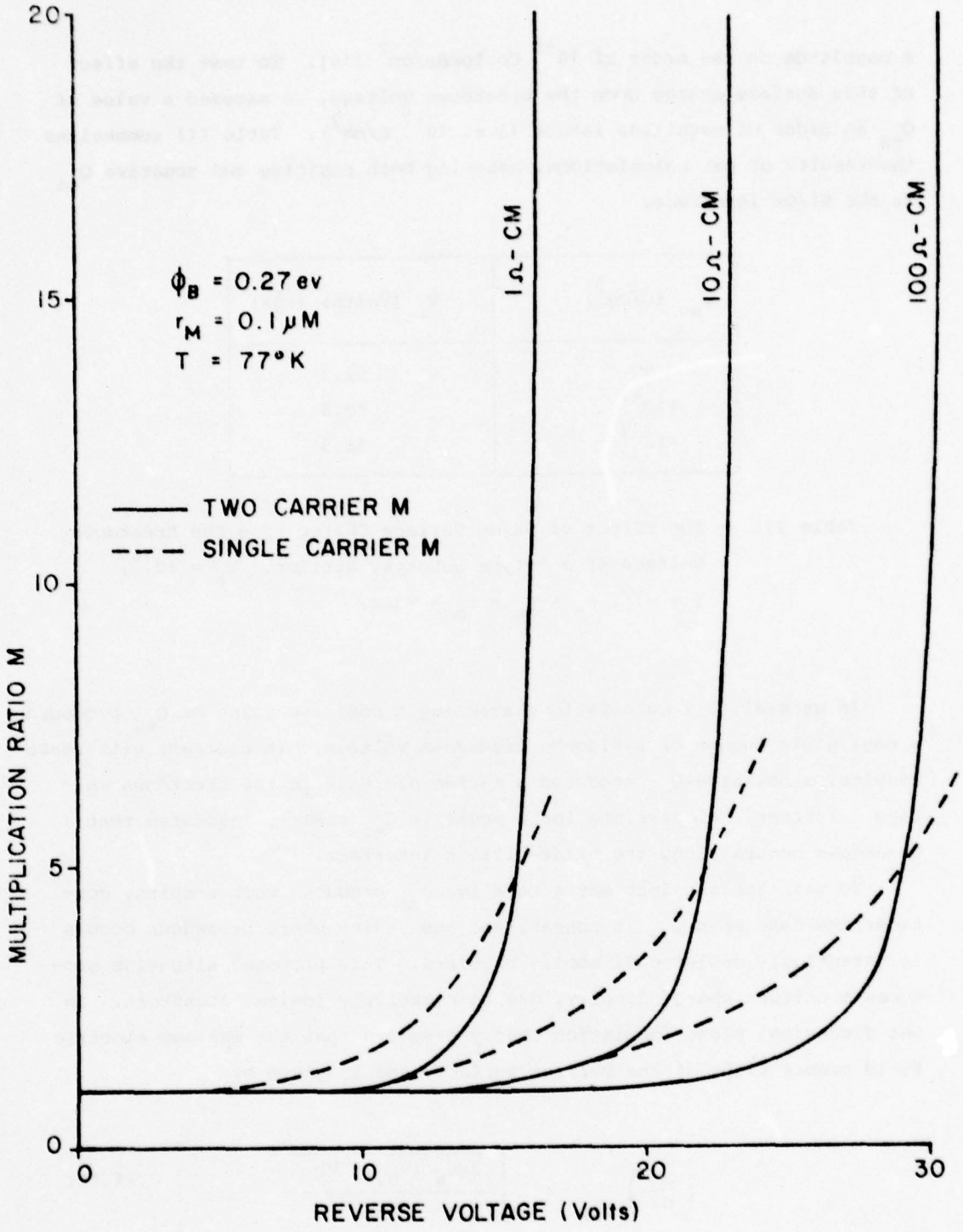


Fig. 7
Multiplication Ratio vs. Applied Reverse Voltage
for a Range of Substrate Resistivities,
r_m = 0.1 um

a magnitude on the order of 10^{-8} Coulombs/cm^2 [16]. To test the effect of this surface charge upon the breakdown voltage, we assumed a value of Q_{ss} an order of magnitude larger (i.e. 10^{-7} C/cm^2). Table III summarizes the results of our calculations, assuming both positive and negative Q_{ss} at the Si/Ox interface.

Q_{ss} (C/cm^2)	V_b (Volts) ($\pm$5%)
0	52.2
$+10^{-7}$	49.8
-10^{-7}	32.3

Table III -- The Effect of Oxide Surface Charge upon the Breakdown Voltage of a P-type Schottky Barrier. $N_a = 10^{16}$, $T = 77^{\circ}$, $x_m = y_m = r_m = 1\,\mu m$.

In general, all calculations assuming a positive value of Q_{ss} produced a negligible change of avalanche breakdown voltage. In contrast with these results, a negative Q_{ss} produced a marked decrease in the breakdown voltage. Further, calculations for a negative Q_{ss} clearly indicated that breakdown occurs along the oxide-silicon interface.

To gain insight into why a negative Q_{ss} produced such results, consider the case when Q_{ss} is absent, and the region where breakdown occurs is essentially depleted of mobile carriers. This proposed situation produces a uniform charge density, due to negatively ionized acceptors. In one dimension, planar depletion theory predicts that the maximum electric field occurs right at the barrier surface, and is given by:

$$\left(\frac{d\psi}{dx}\right)_{max} = \sqrt{\frac{2qN_a\,(V_{bi} + V)}{\epsilon}} \qquad (1.18)$$

Note that the higher the negative charge N_a, the higher the electric field, and therefore the lower the expected breakdown voltage.

Although the planar solution cited cannot predict quantitatively the effect of a surface charge Q_{ss}, it does predict qualitatively the effect observed in Table III. Locally, a negative Q_{ss} would augment the negative charge, due to background acceptors, and a local increase in electric field along the surface. Thereby, from a qualitative point of view, it is suggested that a decrease of breakdown voltage will be observed in the vicinity of a negative surface charge. Conversely, a positive Q_{ss} would tend, locally, to compensate the negative background charge and, thereby, lead to a decrease in electric field, and an associated increase of breakdown voltage. This is generally consistent with our observations in the computer model. Assuming Q_{ss} positive, breakdown was found to occur well away from the SiO_2-Si interface. Conversely, with Q_{ss} negative, it occurred right at that interface.

From the foregoing calculations it is suggested that an oxide surface charge Q_{ss} (which is physically positive) will have a marked effect on the avalanche breakdown voltage of n-type Schottky barriers, and a negligible effect upon p-type structures.

5.0 The Effect of Temperature

Figure 8 presents the calculated breakdown voltage for two-carrier multiplication, as a function of temperature T, for the case $N_a = 1.6 \times 10^{15}$, $\phi_B = 0.27$ ev, and $r_m = 0.1 \mu m$. As previously stated, the computer model utilizes Crowell and Sze's empirical fit to Baraff's curves to estimate the Townsend ionization coefficients α_n and α_p at reduced temperatures. We find that this empirical fit does not adequately agree with published experimental data at temperatures well below $300^{\circ}K$. For this reason, there remains many questions as to whether the specific shape of the curve in Figure 8 is more than qualitatively correct.

From Table I, however, it is clear that a p-type platinum-silicide Schottky barrier goes ohmic at temperatures approaching room temperature. This situation implies that large reverse currents will be observed at voltages well below the $\approx$ 22 volt value shown in Figure 8. For this reason, it appears physically meaningless to undertake breakdown calculations at temperatures such that the "built-in" potential V_{bi} is on the order of a few (kT/q), or less. Mechanisms other than avalanche breakdown come into play at elevated temperatures, and these mechanisms produce

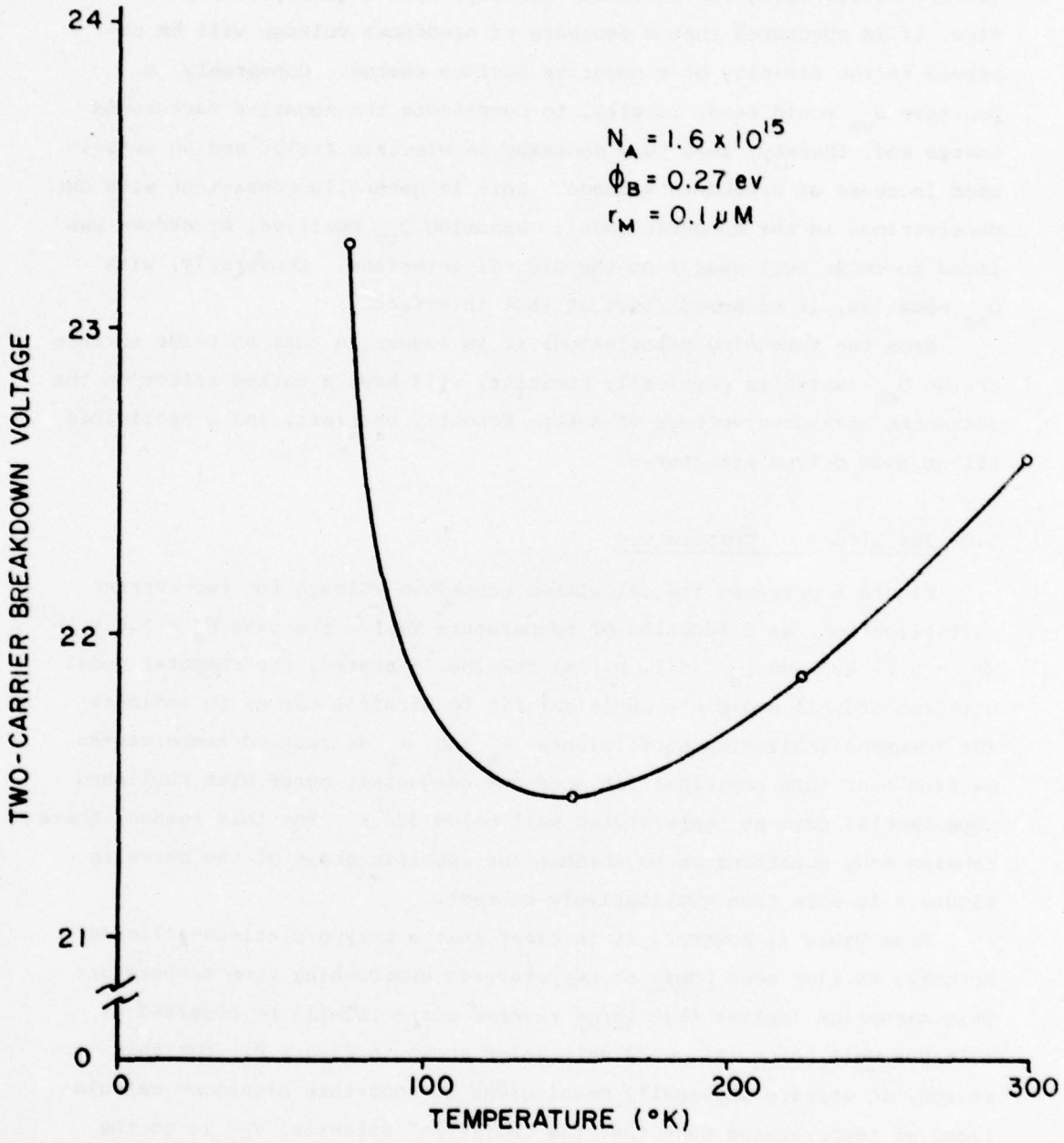


Fig. 8
Calculated Breakdown Voltage for a Two-Carrier Multiplication Process vs. Junction Temperature

large reverse currents at reverse voltages well below those required for significant carrier multiplication.

CHAPTER III

Instructions for Using the Computer Program

1.0 Input Data

Input data are specified in the form of NAMELIST data sets: GEOM, PRMTRS, CONTRL, SWITCH, and SINGLE. Table IV shows a sample set of such input data.

```
&GEOM  XWIDTH=0.10,YDEPTH=0.10,IMAX=31,JMAX=31,NXWDTH=17,NYDPTH=17,
XDELTA=1.00,YDELTA=1.00,RAD=0.10,&END
&PRMTRS  ACCPTR=1.6E15,BRRIER=0.27,QSS=0.00E-00,TMPTR=077.0,&END
&CONTRL  ITMAX=25,ITRMAX=25,CONVRG=0.02,SIDLT=0.005,AVDLT=.01,
VLTDLT=.05,KNTMAX=100,&END
&SINGLE VOLTO=5.0,DLTVLT=5.0,NVOLT=5,KOUTPT=1,&END
&SWITCH NDOPE=0,&END
                              SCHOTTKY BARRIER--SOFT BREAKDOWN CASE
                                   QSS=0.E-00   RAD=.10
```

Table IV -- Sample Input Data

Note that the last 3 cards of the data set are for a 3-line title; thereby a user may identify the purpose of any particular run. It is emphasized that these 3 title cards must be present in the data deck, even if they are left blank. Immediately following the input of these data, the title is printed, in addition to a listing of all input data.

As discussed in Chapter I, two versions of the program exist, the first version iterates the applied voltage to ascertain the applied voltage leading to breakdown, the second version increments the applied voltage over a specified range, and at specified increments. The latter are specified in NAMELIST/SINGLE/. In the event that the first version of the program is being used, NAMELIST/SINGLE/ should be omitted. Failure to do so will result in an EXIT.

It is also emphasized that the order in which these NAMELISTS appear in the data deck must be retained.

1.01 NAMELIST/GEOM/

This data set specifies the geometry to be treated, and it includes:

XWIDTH -- Width of the metal-silicide region in microns (x_m of Figure 1).

YDEPTH -- Depth of the metal-silicide region in microns (y_m of Figure 1).

IMAX -- Total number of lattice points in the x-direction of Figure 1 (31 maximum).

JMAX -- Total number of lattice points in the y-direction of Figure 1 (31 maximum).

NXWDTH -- Number of lattice points in the x-direction which overlay the metal-silicide region (a fraction of IMAX).

NYDPTH -- Number of lattice points in the y-direction which overlay the metal-silicide region (a fraction of JMAX).

XDELTA -- The parameter δ_x (≤ 1) described below.

YDELTA -- The parameter δ_y (≤ 1) described below.

RAD -- Radius of curvature of the metal-silicide corner, in microns (r_m of Figure 1).

The parameters δ_x and δ_y, represented by XDELTA and YDELTA above, control the grading of the lattice illustrated in Figure 2. Both XDELTA and YDELTA must be assigned numerical values ≤ 1. (In generating the graded lattice of Figure 2, XDELTA and YDELTA were equal, with a magnitude of 0.85). These grading parameters function in the following manner.

Let Δx be the interval between two horizontal lattice points located at I and I-1 (where $2 \leq I \leq$ NXWDTH-1). The interval $\Delta x'$ between the lattice points located at I and I + 1 is given by the algorithm $\Delta x\,(I + 1) = \delta_x\, \Delta x(I)$. Thus, with $\delta_x < 1$, the spacing between adjacent lattice points gradually reduces to smaller and smaller values as the index I increments from 2 to NXWDTH-1. The parameter δ_y plays a similar role in grading the vertical lattice spacings within the rectangle which overlays the metal-silicide region. Note that setting both δ_x and δ_y equal to unity leads to a uniform lattice spacing in that region, which is really

the best choice when the geometry treated has circular symmetry (i.e. $x_m = y_m = r_m$ in Figure 1).

It will be noted that the parameters XDELTA and YDELTA are used only for the rectangle which overlays the metal-silicide region. The grading of the remaining lattice spacings (for NXWDTH < I < IMAX and NYDPTH < J < JMAX) is achieved automatically, once the lattice for the metal-silicide region has been determined. Multipliers analogous to XDELTA and YDELTA are determined in order to apportion the remaining lattice points throughout a distance of 1.25 depletion widths, as shown.

1.02 NAMELIST/PRMTRS/

ACCPTR -- The acceptor density N_a (cm^{-3}). (To treat n-type material, this parameter must be assigned the numerical value of the donor density.)

BRRIER -- The Schottky barrier height ϕ_B (ev).

QSS -- The oxide interface interface charge Q_{ss} (C/cm^2).

TMPTR -- The operating temperature T (^oK).

1.03 NAMELIST/CONTRL/

This data set specifies parameters which control the iterative procedures used within the computer model.

ITMAX -- Maximum number of iterations to be allowed in the implementation of Gummel's algorithm. The program will stop execution, and a termination message will be printed if ITMAX is exceeded.

ITRMAX -- Maximum number of iterations to be allowed in the implementation of Stone's method. The program will stop execution, and a termination message will be printed if ITRMAX is exceeded.

CONVRG -- Convergence criterion for Gummel's algorithm.

$$\text{MAX} \left| \psi_m - \psi_{m+1} \right| < \text{CONVRG}$$, where m is the iteration number

SIDLT -- Convergence criterion for Stone's method. When the maximum of the absolute residual (see Stone's paper for its definition) is found less than SIDLT, Stone's method is taken to have converged.

AVDLT -- Convergence criterion for the method of secants in SUBROUTINE AVLNCH. When the denominator in eq. (1.07) is found to have an absolute value less than AVDLT, convergence is satisfied.

VLTDLT -- Convergence criterion for the outer iterative procedure used in SUBROUTINE AVLNCH. When the new value of V is within VLTDLT x 100% of the previous value, convergence is satisfied.

KNTMAX -- Maximum number of points allowed per trajectory in SUBROUTINE AVLNCH. If this is exceeded, execution will stop, and a termination message will be printed, explaining any necessary corrective action.

The sample input data contained in Table IV give values for the above parameters which have been found to work satisfactorily.

1.04 NAMELIST/SINGLE/

This data set is required only when the second version of the program is used. These parameters are designed to step the applied voltage through a prescribed range, listing M (the carrier multiplication) for both two-carrier and single-carrier processes. This data set should be omitted if the first version of this program is to be used.

VOLT ϕ -- The initial (lowest) voltage to be used (volts).

DLTVLT -- The voltage increment (volts) for stepping the applied voltage upwards.

NVOLT -- The number of applied voltage values to be treated.

KOUTPT -- A switching parameter used to bypass the printing of voluminous output of two-dimensional tables. Set KOUTPT = ϕ if detailed data printout is desired. Set KOUTPT = 1 if these data are not required.

1.05 NAMELIST/SWITCH/

This data "set" contains a single parameter used to flag whether p-type or n-type material is to be treated.

NDOPE -- Set NDOPE = ϕ if the substrate is to be considered p-type, or NDOPE = 1 if the substrate is n-type.

2.0 Output Data

The output data printed with the computer model will first be discussed in the context of the second version of the computer program, wherein the

applied reverse voltage is incremented in pre-specified steps. Alterations of the data format for the other version will then be discussed separately.

2.01 Output Data for the Version Which Increments the Applied Voltage in Prespecified Steps

Table V (2 sheets) illustrates the output obtained with the former version. Note that the title that was included with the input data is first printed, followed by a summary of the data specified in the five input NAMELIST sets. This is followed by a summary of the equilibrium parameters at the specified temperature T (77^{o}K in this case). These data are:

TMPTR -- The temperature (^{o}K)

EGAP -- The energy gap Eg (ev)

EIV -- The energy difference ($E_i - E_v$) of Figure 4

EFV -- The energy difference ($E_f - E_v$) of Figure 4

TRNSIC -- The intrinsic carrier density n_i (cm^{-3}) at temperature T.

BKGRND -- The background density of ionized impurities (cm^{-3}) at that temperature

VBLTIN -- The "built-in" potential (volts) of Figure 4

Following the above listing, information is given allowing one to follow the progress of the iterative procedure. The case illustrated possesses circular symmetry ($x_m = y_m = r_m$) and therefore the initial guess for ψ obtained from cylindrical coordinate depletion theory is a reasonable approximation. In this example three iterations of Stone's method were required, and a single application of Gummel's algorithm was sufficient to achieve convergence.

Subsequent to this convergence of the iterative procedure for each voltage step the following data are tabulated:

K -- K = 2 signifies that the central trajectory of the triplet of trajectories normally investigated is the one for which the data are listed. (Actually, because of the circular symmetry of the case illustrated, it was the only one treated.)

SCHOTTKY BARRIER--SHOT BREAKDOWN CASE
VSS=0.E-00 NAD=.10
A.D. SUTHERLAND 10/31/78

SUMMARY OF INPUT DATA :

NAMELIST/GEOM/ XWIDTH= 1.00E-01 YDEPTH= 1.00E-01 IMAX= 51 JMAX= 51 NXHDTH= 17 NYDPTH= 17 XDELTA= 1.00E+00
YDELTA= 1.00E+00 RAD= 1.00E-01

NAMELIST/PHYSKS/ ACCPTR= 1.40E+15 BBWIER= 2.70E-01 VSS= 0.0 TMPTR= 7.70E+01

NAMELIST/CONTRL/ ITMAX= 25 ITMMAX= 25 CONVRG= 2.00E-02 SIDLT= 5.00E-03
AVDLT= 1.00E-02 VLTDLT= 5.00E-02 KNTMAX= 100

NAMELIST/SINGLE/ VOLTO= 5.00E+00 DLTVLT= 5.00E+00 NVOLT= 5 KOUTPT= 1

NAMELIST/SWITCH/ NOPPR= 0

TMPTR	EGAP	EIV	EIF	EFV	TWNSIC	BARMND	VBLTIN
7.70E+01	1.19E+00	5.77E-01	5.24E-01	5.32E-02	5.24E-20	1.02E+15	2.17E-01

3 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
MAX DELTA IN GRELAX IS -5.586E-03 AT COORDINATE (50,51)
1 ITERATIONS WERE REQUIRED IN GRELAX

K	ISTART	JSTART	XPATH	YPATH	ET	ALPHN	ALPHP	PI	OMEGA
2	12	20	7.07E-02	7.07E-02	2.65E+05				
			7.12E-02	7.82E-02	2.00E+05			0.0	0.0
			8.48E-02	8.48E-02	1.87E+05	1.29E+03	5.38E+00	2.67E-03	2.19E-03
			9.10E-02	9.10E-02	1.75E+05	5.24E+02	3.55E-01	3.04E-03	2.45E-03
			9.68E-02	9.68E-02	1.64E+05	1.38E+02	5.78E-02	3.14E-03	2.24E-03
			1.02E-01	1.02E-01	1.55E+05	5.40E+01	8.48E-03	3.25E-03	2.24E-03
						2.10E+01	1.21E-03	3.27E-03	2.24E-03

VOLT= 5.00E+00 n1= 1.00E+00 n2= 1.00E+00

3 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
MAX DELTA IN GRELAX IS -9.340E-03 AT COORDINATE (50,51)
1 ITERATIONS WERE REQUIRED IN GRELAX

K	ISTART	JSTART	XPATH	YPATH	ET	ALPHN	ALPHP	PI	OMEGA
2	12	20	7.07E-02	7.07E-02	4.59E+05				
			8.11E-02	8.11E-02	3.48E+05			0.0	0.0
			9.04E-02	9.04E-02	3.16E+05	4.62E+04	5.02E+03	8.44E-02	1.58E-02
			9.89E-02	9.89E-02	2.84E+05	2.36E+04	1.95E+03	1.29E-01	1.87E-02
			1.08E-01	1.07E-01	2.63E+05	1.36E+04	5.72E+02	1.44E-01	1.97E-02
			1.18E-01	1.14E-01	2.45E+05	7.79E+03	1.89E+02	1.53E-01	2.00E-02
			1.60E-01	1.20E-01	2.31E+05	4.55E+03	6.59E+01	1.58E-01	2.01E-02
			1.67E-01	1.27E-01	2.19E+05	2.72E+03	2.39E+01	1.61E-01	2.01E-02
			1.35E-01	1.33E-01	1.99E+05	1.65E+03	8.61E+00	1.63E-01	2.01E-02
			1.49E-01	1.39E-01	1.84E+05	7.22E+02	1.67E+00	1.64E-01	2.01E-02
						2.81E+02	2.47E-01	1.65E-01	2.01E-02

VOLT= 1.00E+01 n1= 1.18E+00 n2= 1.02E+00

3 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN

TABLE V
Sample Output Data
(V_{app} stepped)

1 ITERATIONS WERE REQUIRED IN GRELAX

K	ISTART	JSTART	XPATH	YPATH	ET	ALPHN	ALPHP	PI	UMEGA
2	12	20	7.07E-02	7.07E-02	6.50E+05			0.0	0.0
			8.54E-02	8.34E-02	4.79E+05	1.37E+03	4.81E+04	2.93E-01	1.64E-01
			9.49E-02	9.45E-02	4.25E+05	8.35E+04	1.84E+04	4.34E-01	1.70E-01
			1.09E-01	1.09E-01	3.78E+05	5.48E+04	8.19E+03	5.24E-01	1.90E-01
			1.13E-01	1.19E-01	3.44E+05	3.67E+04	5.84E+03	5.75E-01	1.99E-01
			1.22E-01	1.21E-01	3.21E+05	2.52E+04	1.90E+03	6.07E-01	2.01E-01
			1.29E-01	1.24E-01	3.01E+05	1.78E+04	4.44E+02	6.24E-01	2.04E-01
			1.36E-01	1.32E-01	2.85E+05	1.24E+04	4.80E+02	6.42E-01	2.05E-01
			1.50E-01	1.50E-01	2.53E+05	7.00E+03	1.35E+02	6.57E-01	2.04E-01
			1.62E-01	1.62E-01	2.38E+05	5.65E+03	4.28E+01	6.65E-01	2.04E-01
			1.74E-01	1.71E-01	2.22E+05	1.43E+03	1.21E+01	6.84E-01	2.04E-01
			1.80E-01	1.83E-01	2.08E+05	1.05E+03	3.53E+00	6.70E-01	2.06E-01

VOLT= 1.30E+01 M1= 1.95E+00 M2= 1.26E+00

3 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN

MAX DELTA IN GRELAX IS -1.142E-02 AT COORDINATE (50,51)

1 ITERATIONS WERE REQUIRED IN GRELAX

K	ISTART	JSTART	XPATH	YPATH	ET	ALPHN	ALPHP	PI	UMEGA
2	12	20	7.07E-02	7.07E-02	7.97E+05			0.0	0.0
			8.54E-02	8.54E-02	6.09E+05	2.51E+03	1.19E+05	5.45E-01	5.50E-01
			9.80E-02	9.80E-02	5.20E+05	1.54E+03	5.71E+04	8.44E-01	5.15E-01
			1.09E-01	1.09E-01	4.61E+05	1.09E+03	3.05E+04	1.03E+00	6.03E-01
			1.19E-01	1.19E-01	4.20E+05	7.94E+04	1.68E+04	1.13E+00	6.54E-01
			1.28E-01	1.28E-01	3.89E+05	5.90E+04	9.50E+03	1.23E+00	6.84E-01
			1.36E-01	1.36E-01	3.69E+05	4.65E+04	5.82E+03	1.29E+00	6.94E-01
			1.44E-01	1.44E-01	3.49E+05	3.59E+04	3.36E+03	1.35E+00	7.07E-01
			1.59E-01	1.59E-01	3.10E+05	2.17E+04	1.41E+03	1.39E+00	7.13E-01
			1.73E-01	1.72E-01	2.85E+05	1.31E+04	5.28E+02	1.41E+00	7.14E-01
			1.85E-01	1.84E-01	2.65E+05	9.04E+03	2.08E+02	1.43E+00	7.20E-01
			1.97E-01	1.98E-01	2.49E+05	5.02E+03	9.08E+01	1.43E+00	7.20E-01
			2.18E-01	2.17E-01	2.24E+03	2.29E+03	1.89E+01	1.44E+00	7.20E-01
			2.37E-01	2.39E-01	2.05E+05	9.45E+02	2.86E+00	1.45E+00	7.20E-01

VOLT= 2.00E+01 M1= 4.25E+00 M2= 3.57E+00

3 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN

MAX DELTA IN GRELAX IS -1.038E-02 AT COORDINATE (50,51)

1 ITERATIONS WERE REQUIRED IN GRELAX

K	ISTART	JSTART	XPATH	YPATH	ET	ALPHN	ALPHP	PI	UMEGA
2	12	20	7.07E-02	7.07E-02	9.61E+05			0.0	0.0
			8.71E-02	8.71E-02	7.13E+05	5.12E+03	2.04E+05	8.04E-01	5.51E-01
			1.01E-01	1.01E-01	6.08E+05	2.19E+03	1.09E+05	1.27E+00	9.77E-01
			1.13E-01	1.13E-01	5.39E+05	1.64E+03	6.41E+04	1.56E+00	1.14E+00
			1.23E-01	1.23E-01	4.87E+05	1.26E+03	3.98E+04	1.74E+00	1.32E+00
			1.33E-01	1.33E-01	4.51E+05	9.84E+04	2.31E+04	1.91E+00	1.41E+00
			1.42E-01	1.42E-01	4.21E+05	7.80E+04	1.62E+04	2.01E+00	1.47E+00
			1.51E-01	1.50E-01	3.96E+05	6.22E+04	1.04E+04	2.09E+00	1.50E+00
			1.67E-01	1.65E-01	3.57E+05	4.31E+04	5.27E+03	2.20E+00	1.55E+00
			1.81E-01	1.80E-01	3.28E+05	2.84E+04	2.54E+03	2.28E+00	1.58E+00
			1.94E-01	1.93E-01	3.05E+05	1.91E+04	1.11E+03	2.30E+00	1.57E+00
			2.07E-01	2.05E-01	2.86E+05	1.30E+04	5.24E+02	2.33E+00	1.58E+00
			2.27E-01	2.28E-01	2.57E+05	6.66E+03	1.89E+02	2.35E+00	1.58E+00
			2.50E-01	2.44E-01	2.35E+05	3.36E+03	3.43E+01	2.37E+00	1.58E+00
			2.59E-01	2.67E-01	2.18E+05	1.67E+03	8.48E+00	2.37E+00	1.58E+00

VOLT= 2.50E+01 M1= 1.07E+01 M2= -1.73E+00

IMSL001 EXECUTION TERMINATING DUE TO ERROR COUNT FOR ERROR NUMBER 217

TRACE BACK = ERR OR DATA SET NOT LOST

I START -- The indices (I, J) of the lattice point at the upper left corner
J START of the "box" containing the starting location of the trajectory.

XPATH -- The (x, y) coordinates (in microns) of the charge trajectory as
YPATH time of flight increases.

ET -- The total electric field ($E_t = \sqrt{E_x^2 + E_y^2}$) (in V/cm) at that location.

ALPHN -- The Townsend coefficients α_p, α_p, obtained from Crowell and
ALPHP Sze's empirical fit to Baraff's curves (cm^{-1}).

PI -- The value of the integral in eq. (1.10) for the care of single-carrier multiplication.

OMEGA -- The value of the integral in eq. (1.07) for the care of two-carrier multiplication.

Note that the two integrals converge to a constant value as the charge trajectory extends into regions of small electric field. The numerical integration terminates when the most recent value for Ω is within .01% of its previous value.

Following the above tabulations are printed the parameters:

VOLT -- The applied voltage V (volts)

M1 -- The single-carrier multiplication ratio given by eq. (1.10).

M2 -- The two-carrier multiplication ratio given by eq. (1.07).

For the case illustrated, note that the latter changes sign (because the integral Ω exceeds unity) with V = 25 volts. This signifies that avalanche breakdown occurred at a voltage somewhere between 20 and 25 volts, for this case. (Figure 5 shows it to occur at V = 22.6 volts.)

The case illustrated in Table V was run with KOUTPUT = 1, thereby bypassing the printing of additional output data. Had KOUTPT = 0, however, additional tables of data showing would have been listed. These tables are all formatted in an identical manner, so it suffices to illustrate and discuss only one of them.

Table VI (2 sheets) illustrates the data listed for the electrostatic potential, in unnormalized units. These data were generated for a case differing from the one discussed above. One reads the table in the following manner. Listed across the top of the first sheet are the x-coordinates X(I) of the lattice, in microns. Just below that are the headings "J, Y(J),

TABLE IV

Electrostatic Potential in Un-normalized Units

X(I), I=1, IMAX

0.0	1.00E-02	2.00E-02	3.00E-02	4.00E-02	5.00E-02	6.00E-02	7.00E-02	8.00E-02	9.00E-02
1.00E-01	1.11E-01	1.23E-01	1.40E-01	1.63E-01	1.92E-01	2.30E-01	2.81E-01	3.47E-01	4.33E-01
5.44E-01	6.93E-01	8.87E-01	1.14E+00	1.47E+00	1.90E+00	2.47E+00	3.21E+00	4.19E+00	5.44E+00
7.17E+00									

J	Y(J)	PST(I,J), I=1, IMAX									
31	0.0	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01
		2.49E+01	2.40E+01	2.31E+01	2.19E+01	2.07E+01	1.92E+01	1.74E+01	1.60E+01	1.42E+01	1.24E+01
		1.04E+01	8.81E+00	6.88E+00	4.90E+00	3.15E+00	1.43E+00	3.61E-01	2.80E-03	1.98E-04	0.0
		0.0									
30	1.00E-02	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01
		2.48E+01	2.40E+01	2.30E+01	2.19E+01	2.06E+01	1.92E+01	1.74E+01	1.60E+01	1.42E+01	1.24E+01
		1.04E+01	8.81E+00	6.88E+00	4.90E+00	3.15E+00	1.43E+00	3.61E-01	2.80E-03	2.01E-04	0.0
		0.0									
29	2.00E-02	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01
		2.47E+01	2.39E+01	2.30E+01	2.19E+01	2.06E+01	1.92E+01	1.74E+01	1.60E+01	1.42E+01	1.24E+01
		1.04E+01	8.81E+00	6.87E+00	4.90E+00	3.15E+00	1.43E+00	3.60E-01	2.80E-03	2.04E-04	0.0
		0.0									
28	3.00E-02	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01
		2.45E+01	2.37E+01	2.28E+01	2.18E+01	2.05E+01	1.91E+01	1.74E+01	1.59E+01	1.42E+01	1.24E+01
		1.04E+01	8.80E+00	6.87E+00	4.90E+00	3.15E+00	1.43E+00	3.60E-01	2.80E-03	2.20E-04	0.0
		0.0									
27	4.00E-02	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01
		2.42E+01	2.35E+01	2.26E+01	2.16E+01	2.04E+01	1.90E+01	1.75E+01	1.59E+01	1.42E+01	1.24E+01
		1.04E+01	8.80E+00	6.87E+00	4.90E+00	3.15E+00	1.43E+00	3.60E-01	2.80E-03	2.37E-04	0.0
		0.0									
26	5.00E-02	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.44E+01
		2.39E+01	2.32E+01	2.24E+01	2.14E+01	2.02E+01	1.89E+01	1.74E+01	1.58E+01	1.42E+01	1.24E+01
		1.04E+01	8.79E+00	6.86E+00	4.89E+00	3.15E+00	1.43E+00	3.60E-01	2.80E-03	2.60E-04	0.0
		0.0									
25	6.00E-02	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.42E+01
		2.35E+01	2.29E+01	2.21E+01	2.12E+01	2.01E+01	1.88E+01	1.74E+01	1.58E+01	1.41E+01	1.24E+01
		1.04E+01	8.74E+00	6.84E+00	4.89E+00	3.14E+00	1.43E+00	3.60E-01	2.80E-03	2.89E-04	0.0
		0.0									
24	7.00E-02	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.43E+01	2.37E+01
		2.32E+01	2.26E+01	2.18E+01	2.10E+01	1.99E+01	1.87E+01	1.73E+01	1.57E+01	1.41E+01	1.23E+01
		1.04E+01	8.77E+00	6.85E+00	4.89E+00	3.14E+00	1.43E+00	3.60E-01	2.81E-03	3.25E-04	0.0
		0.0									
23	8.00E-02	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.47E+01	2.38E+01	2.33E+01
		2.27E+01	2.22E+01	2.15E+01	2.07E+01	1.97E+01	1.85E+01	1.72E+01	1.57E+01	1.40E+01	1.23E+01
		1.04E+01	8.74E+00	6.84E+00	4.88E+00	3.14E+00	1.42E+00	3.60E-01	2.81E-03	3.69E-04	0.0
		0.0									
22	9.00E-02	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.49E+01	2.46E+01	2.42E+01	2.37E+01	2.33E+01	2.28E+01
		2.21E+01	2.14E+01	2.12E+01	2.06E+01	1.95E+01	1.83E+01	1.70E+01	1.54E+01	1.40E+01	1.23E+01
		1.05E+01	8.75E+00	6.84E+00	4.88E+00	3.14E+00	1.42E+00	3.59E-01	2.82E-03	4.22E-04	0.0
		0.0									
21	1.00E-01	2.49E+01	2.48E+01	2.47E+01	2.45E+01	2.42E+01	2.39E+01	2.35E+01	2.32E+01	2.27E+01	2.23E+01
		2.14E+01	2.14E+01	2.04E+01	2.01E+01	1.92E+01	1.82E+01	1.69E+01	1.55E+01	1.39E+01	1.22E+01
		1.05E+01	8.74E+00	6.83E+00	4.87E+00	3.13E+00	1.42E+00	3.59E-01	2.83E-03	4.85E-04	0.0
		0.0									
20	1.10E-01	2.48E+01	2.40E+01	2.39E+01	2.37E+01	2.35E+01	2.32E+01	2.28E+01	2.24E+01	2.22E+01	2.18E+01
		2.14E+01	2.10E+01	2.05E+01	1.98E+01	1.90E+01	1.80E+01	1.68E+01	1.54E+01	1.39E+01	1.22E+01
		1.05E+01	8.72E+00	6.82E+00	4.86E+00	3.13E+00	1.42E+00	3.59E-01	2.84E-03	5.59E-04	0.0
		0.0									
19	1.23E-01	2.31E+01	2.30E+01	2.30E+01	2.28E+01	2.24E+01	2.24E+01	2.21E+01	2.18E+01	2.15E+01	2.12E+01
		2.04E+01	2.05E+01	2.00E+01	1.94E+01	1.87E+01	1.77E+01	1.64E+01	1.52E+01	1.38E+01	1.22E+01
		1.05E+01	8.70E+00	6.80E+00	4.85E+00	3.12E+00	1.42E+00	3.59E-01	2.84E-03	6.74E-04	0.0
		0.0									
18	1.40E-01	2.19E+01	2.19E+01	2.18E+01	2.17E+01	2.14E+01	2.14E+01	2.12E+01	2.09E+01	2.07E+01	2.04E+01
		2.01E+01	1.94E+01	1.94E+01	1.89E+01	1.82E+01	1.72E+01	1.63E+01	1.51E+01	1.36E+01	1.21E+01
		1.04E+01	8.67E+00	6.78E+00	4.84E+00	3.12E+00	1.42E+00	3.58E-01	2.84E-03	8.71E-04	0.0
		0.0									
17	1.47E-01	2.04E+01	2.04E+01	2.04E+01	2.05E+01	2.04E+01	2.02E+01	2.01E+01	1.99E+01	1.97E+01	1.94E+01

and PSI(I, J), I = 1, IMAX", indicating that each subsequent grouping lists the potential along a particular row of the lattice (whose index is J and whose y-coordinate is Y(J), in microns). Thus, first scanning the listing of X(I), one then locates the corresponding potential and the y-coordinate Y(J) in the grouping labeled J. In this manner, the two-dimensional solutions may be determined.

2.02 Output Data for the Version Which Iterates the Applied Voltage to Determine Avalanche Breakdown

Table VII (3 sheets) illustrates the output data printed for the alternate version of the computer program. The initial printout, including the equilibrium data, remains the same as that of the previous subsection. Following that is listed the initial voltage estimate (obtained from SUBROUTINE VGUESS) with which the iterative procedure starts. Next follows the summary of the iterative process which, in this case, requires more iterations with both Stone's method and Gummel's algorithm. This is because this version of the program uses planar depletion theory to supply an initial estimate for potential--an estimate which decreases in accuracy, with a decrease in radius of curvature.

With the solution of Poisson's equation now known, the program next deals with triplets of trajectories, labeled K = 1, 2, 3, respectively, as it searches for the trajectory producing the maximum value of the integral appearing in equation (1.07). Normally, the next sequence of data printed would be a multiple set similar to that typified by the single set seen following the iteration summary in Table VII, detailing what is taking place in this search process. For the case illustrated, however, the problem has circular symmetry, and the print-out is unnecessary. Thus, only a single subset of summary data appears in the table, for the trajectory K = 2.

The data included in this subset are:

K -- The label assigned to the trajectory

ISTART -- The starting indices of the trajectory
JSTART

XSTART -- The starting coordinates (x, y) of the trajectory, in microns.
YSTART

```
                    SCHOTTKY BARRIER--SHALLOW-- 77 DEG K
                              QSS=0
                    A.D.SUTHERLAND--4,20,78

 SUMMARY   OF INPUT DATA :

  NAMELIST/GEOM/ XWIDTH= 1.00E-01  YDEPTH= 1.00E-01  IMAX= 31  JMAX= 31  NXWDTH= 11  NYDPTH= 11  XDELTA= 1.00E+00
  YDELTA= 1.00E+00  PAR= 1.00E-01

  NAMELIST/PRMTRS/ ACCPTR= 1.60E+15  BRRIER= 2.70E-01  QSS= 0.0       THPTR= 7.70E+01

  NAMELIST/CONTRL/ ITMAX= 25  ITRMAX= 25  CONVRG= 2.00E-02  SIDLT= 5.00E-03
  AVOLT= 1.00E-02  VLTDLT= 5.00E-02  KNTMAX= 100

 THPTR     FGAP      EIV       EIF       EFV       TRNSIC    BKGRND    VBLTIN
 7.70E+01  1.16E+00  5.77E-01  5.24E-01  5.32E-02  5.24E-20  1.02E+15  2.17E-01

 THE INITIAL VOLTAGE ESTIMATE IS  2.44E+01 VOLTS

  6 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
  MAX DELTA IN GRELAX IS -5.453E-01 AT COORDINATE (19, 8)

  5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
  MAX DELTA IN GRELAX IS  6.375E-02 AT COORDINATE (16,14)

  5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
  MAX DELTA IN GRELAX IS  1.698E-01 AT COORDINATE (22,31)

  5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
  MAX DELTA IN GRELAX IS  8.373E-02 AT COORDINATE (23,11)

  5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
  MAX DELTA IN GRELAX IS  6.327E-02 AT COORDINATE (22,31)

  5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
  MAX DELTA IN GRELAX IS  3.672E-02 AT COORDINATE ( 1, 6)

  5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
  MAX DELTA IN GRELAX IS  2.341E-02 AT COORDINATE (25, 7)

  4 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
  MAX DELTA IN GRELAX IS  2.045E-02 AT COORDINATE (26,31)

  4 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
  MAX DELTA IN GRELAX IS -1.510E-02 AT COORDINATE ( 1,13)
      9  ITERATIONS WERE REQUIRED IN GRELAX

 K   ISTART JSTART   XSTART    YSTART    XPATH     YPATH      FT        ALPHA     PI
```

TABLE VII
Sample Output Data (Avalanche Calculation)

```
2      8     24     7.07E-02  7.07E-02  7.07E-02  7.07E-02  8.77E+05                 0.0
                                         1.27E-01  1.27E-01  4.73E+05  4.62E+04  8.12E-01
                                         1.68E-01  1.68E-01  3.60E+05  1.31E+04  9.23E-01
                                         1.93E-01  1.94E-01  3.00E+05  5.14E+03  9.55E-01
                                         2.14E-01  2.19E-01  2.61E+05  2.31E+03  9.64E-01
                                         2.39E-01  2.42E-01  2.33E+05  1.15E+03  9.71E-01
                                         2.59E-01  2.62E-01  2.13E+05  6.18E+02  9.73E-01
                                         2.77E-01  2.81E-01  1.96E+05  3.47E+02  9.74E-01
                                         3.10E-01  3.15E-01  1.74E+05  1.43E+02  9.75E-01
                                         3.39E-01  3.45E-01  1.57E+05  6.23E+01  9.76E-01
                                         3.64E-01  3.73E-01  1.44E+05  3.14E+01  9.76E-01
                                         3.91E-01  3.98E-01  1.36E+05  1.70E+01  9.76E-01

KEY= 1       PSINRM= 2.702E+01    DSTNRM= 5.857E-04

KNT    DX          DY          ET1         ET2         ALPHA       PI

 2   5.89E-06    5.92E-06    9.19E+05    4.94E+05    5.34E+04    9.55E-01
 3   3.84E-06    3.89E-06    4.94E+05    3.77E+05    1.60E+04    1.10E+00
 4   3.04E-06    3.12E-06    3.77E+05    3.14E+05    6.54E+03    1.14E+00
 5   2.61E-06    2.67E-06    3.14E+05    2.73E+05    3.04E+03    1.15E+00
 6   2.29E-06    2.36E-06    2.77E+05    2.45E+05    1.55E+03    1.14E+00
 7   2.07E-06    2.13E-06    2.45E+05    2.23E+05    8.44E+02    1.16E+00
 8   1.90E-06    1.96E-06    2.23E+05    2.04E+05    4.88E+02    1.16E+00
 9   1.42E-06    1.53E-06    2.04E+05    1.82E+05    2.04E+02    1.17E+00
10   3.07E-06    3.17E-06    1.82E+05    1.65E+05    9.24E+01    1.17E+00
11   2.81E-06    2.91E-06    1.65E+05    1.51E+05    4.75E+01    1.17E+00
12   2.62E-06    2.70E-06    1.51E+05    1.41E+05    2.61E+01    1.17E+00
13   4.78E-06    4.94E-06    1.41E+05    1.24E+05    1.03E+01    1.17E+00

KEY= 2       PSINRM= 2.488E+01    DSTNRM= 5.620E-04

KNT    DX          DY          ET1         ET2         ALPHA       PI

 2   5.65E-06    5.68E-06    8.82E+05    4.74E+05    4.71E+04    8.29E-01
 3   3.69E-06    3.73E-06    4.74E+05    3.62E+05    1.34E+04    9.44E-01
 4   2.94E-06    3.00E-06    3.62E+05    3.02E+05    5.31E+03    9.77E-01
 5   2.50E-06    2.54E-06    3.02E+05    2.62E+05    2.40E+03    9.89E-01
 6   2.20E-06    2.24E-06    2.62E+05    2.35E+05    1.20E+03    9.94E-01
 7   1.99E-06    2.05E-06    2.35E+05    2.14E+05    6.41E+02    9.96E-01
 8   1.82E-06    1.88E-06    2.14E+05    1.98E+05    3.63E+02    9.98E-01
 9   1.28E-06    1.39E-06    1.98E+05    1.75E+05    1.50E+02    9.99E-01
10   2.95E-06    3.04E-06    1.75E+05    1.58E+05    6.57E+01    9.99E-01
11   2.69E-06    2.79E-06    1.58E+05    1.47E+05    3.32E+01    9.99E-01
12   2.52E-06    2.59E-06    1.47E+05    1.37E+05    1.80E+01    9.99E-01

 1 ITERATIONS WERE REQUIRED WITH THE METHOD OF SECANTS IN AVLNCH


   6 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
   MAX DELTA IN GRELAX IS -5.443E-01 AT COORDINATE (19, 8)

   5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
   MAX DELTA IN GRELAX IS  6.347E-02 AT COORDINATE (16,14)

   5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
   MAX DELTA IN GRELAX IS  1.698E-01 AT COORDINATE (22,31)

   5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
   MAX DELTA IN GRELAX IS  8.400E-02 AT COORDINATE (23,11)

   5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
   MAX DELTA IN GRELAX IS  6.328E-02 AT COORDINATE (22,31)

   5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
```

```
MAX DELTA IN GRELAX IS  3.673E-02 AT COORDINATE ( 1, 6)

5 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
MAX DELTA IN GRELAX IS  2.399E-02 AT COORDINATE (25, 7)

4 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
MAX DELTA IN GRELAX IS  2.047E-02 AT COORDINATE (26,31)

4 ITERATIONS WERE REQUIRED WITH STONES METHOD IN POISSN
MAX DELTA IN GRELAX IS -1.516E-02 AT COORDINATE ( 1,13)
    9  ITERATIONS WERE REQUIRED IN GRELAX
```

XPATH -- The coordinates (x, y), in microns, of the moving charge as time
YPATH progresses.

ET -- The total electric field E_t (V/cm) at that location

ALPHA -- The Kennedy-O'Brien value for α .

PI -- The value of the integral in the denominator of (1.07)

With the trajectory maximizing PI determined, the program next modifies the applied voltage iteratively, scaling E_t along the trajectory in the manner described in Chapter I, and respects the evaluation of the integral PI until the most recent two values bracket unity. Again, subsets of summary data are printed during this search for the bracketing condition, so that the process can be monitored. In the case illustrated, this bracketing required only a single step, so only a single subset of such data were printed. Normally, more than one such subset might be expected to appear.

This subset lists the following parameters:

KEY -- a flag; when equal unity indicating iteration subset

PSINRM-- The total voltage (V_{bi} + V) in volts

DSTNRM-- The (planar) depletion width W_d (cm)

KNT -- A counter for the number of trajectory points in the numerical integration

DX -- Incremental distances (cm) along the path of the numerical in-
DY tegration

ET1 -- The electric field E_t at the beginning and end of the integration
ET2 interval

ALPHA -- The Kennedy-O'Brien empirical Townsend coefficient α

PI -- The ionization integral in the denominator of eq. (1.07) with α_n and α_p replaced by α

Upon finding two adjacent trial voltages which cause the ionization integral PI to bracket unity, the secants technique is used to zero in on the voltage which causes PI to equal unity. The flag KEY is set to 2 for this next sequence of steps. Again, a number of subsets of data are

printed, each time the root-finding program goes through one of its iterations. In the case illustrated, only one such subset was printed because the program converged to the desired voltage in a single iteration. The parameters in this subset are the same as those just defined above.

The above data have all been determined using the first solution of Poisson's equation for the initial estimate of the applied voltage V. Now that a revised estimate for this voltage is available an updated solution of Poisson's equation is obtained, and the process is repeated. Again, a summary of the convergence of the iterative determination of the solution of Poisson's equation is listed, etc. In the case illustrated, the revised estimate of the applied voltage required for breakdown was within the 5% convergence range specified by VLTDLT in NAMELIST/CONTRL/, and so the execution was terminated.

Rather than a STOP, when convergence of this process is detected, this version of the program first prints all of the two-dimensional data exemplified by Table VI (not repeated in Table VII), then prints the resultant breakdown voltage. Then the program loops back to the beginning and searches for a complete new set of input data. This allows multiple cases to be treated in one run. Failing to find such additional data, the program exits.

CHAPTER IV

Conclusions

Several different mechanisms could produce enhanced reverse current (or breakdown) in a Pt-Si Schottky barrier diode. In two spatial dimensions, a small radius of curvature at the junction periphery would locally increase the space-charge layer electric field and, hence, increase carrier multiplication. The existence of surface states at the metal-Si interface (and an associated Q_{ss}) could also enhance the reverse current (or breakdown), although this mechanism should not be significant in p-type devices. Similarly, lattice strain, micro-cracks, or other lattice damage could produce an enhanced reverse current; the presence of these lattice defects appear unlikely to produce a true breakdown situation.

Clearly, there are several potential explanations for the large reverse current exhibited by p-type Pt-Si barriers. Our initial direction was to attribute this reverse current to a classical Townsend avalanche process, and to model this process in two spatial dimensions. After the computer model became operational, substantial qualitative differences were observed between the reverse volt-ampere characteristics of a laboratory device and calculated results. Most laboratory devices exhibit a relatively soft reverse characteristic, whereas the computer model predicted a relatively abrupt reverse breakdown. This comparison between experiment and theory suggested that the reverse current of laboratory devices could be attributable to mechanisms other than avalanche breakdown.

From Townsend's theory of carrier multiplication it is known that one-carrier processes yield a relatively soft breakdown characteristic. Further, other workers have suggested that the ionization rate of electrons at low temperatures is substantially greater than the ionization rate for holes. It is for this reason that a one-carrier multiplication process becomes a possible mechanism to explain the reverse volt-ampere characteristics of these devices. An important question arises concerning the available density of electrons in p-type silicon at $70^{\circ}K$. Calculations for ideal p-type material indicate an electron density (at $70^{\circ}K$) that is too low to produce a significant amount of reverse current.

It is suggested that lattice strain between Pt/Pt-Si/Si is sufficient to produce substantial lattice damage, and a ready source of free electrons. Such lattice damage has been reported in the technical literature. Further, it is suggested that lattice damage arising in devices containing a large quantity of Pt is an important reason why they are of poor electrical quality. Most workers find it necessary to use very small quantities of Pt to successfully fabricate a Pt-Si Schottky barrier.

Assuming lattice damage produces the necessary electrons, a one-carrier multiplication process could explain the soft reverse current characteristics observed in laboratory experiments. Further, theory predicts that one-carrier multiplication processes yield an exponential increase of reverse current, with an increase of reverse voltage. Crude evaluations of practical devices show that an exponential volt-ampere relation is frequently observed over a substantial range of reverse bias. For this reason, one-carrier multiplication could provide a substantial portion of the excessive reverse current in p-type Pt-Si Schottky barrier diodes.

In conjunction with studies unrelated to this effort it was realized that other mechanisms could produce an exponential reverse volt-ampere characteristic. In particular, carrier tunneling at the metal-Si interface could yield an exponential volt-ampere characteristic, similar to that predicted for one-carrier multiplication. A very crude model was developed for tunneling at the Pt-Si interface; this model qualitatively established the suspected volt-ampere relation, although quantitative results were of little value.

From this effort it is concluded that additional work is needed before an adequate model is available for reverse breakdown in a p-type Pt-Si device. At this point reverse current appears attributable to either a one-carrier multiplication or to tunneling at the Pt-Si interface. Before the dominant mechanism becomes identifiable, an accurate quantum mechanical solution should be obtained for tunneling through the Pt-Si interface barrier, as a function of an applied reverse biasing voltage. With such a model, it is suggested that reverse current calculations for a p-type Pt-Si barrier would yield information important to the solution of this fundamental device problem.

References

1. H. Muta and D. Shinoda, "Solid-Solid Reactions in Pt-Si Systems," J. Appl. Phys. 43, 2913 (1972).

2. J. M. Poate, "Kinetics and Mechanism of Platinum Silicide Formation on Silicon," Appl. Phys. Letters, 24, 391 (1974).

3. K. Chino and Y. Mizushima, "Behavior of the Schottky Barrier Diode under Uniaxial Stress," Japanese Jour. Appl. Phys., 13, 156 (1974).

4. H. L. Stone, "Iterative Solution of Implicit Approximations of Multi-Dimensional Partial Differential Equations," SIAM Jour. Numer. Anal., 5, 530 (1968).

5. H. K. Gummel, "A Self-Consistent Iterative Scheme for One-Dimensional Steady State Transistor Calculations," JEEE Trans. Electron Dev., ED-11, 455, 1964.

6. A. D. Sutherland, "A Two-Dimensional Computer Model for the Steady-State Operation of MOSFET's," U. S. Army Electronics Command Research and Development Technical Report, ECOM-75-1344-F Supplement, September, 1977.

7. F. S. Acton, NUMERICAL METHODS THAT WORK, Harper & Rowe, New York, 1970, p. 146.

8. S. M. Sze, PHYSICS OF SEMICONDUCTOR DEVICES, Wiley, New York, 1969, p. 60.

9. D. P. Kennedy and R. R. O'Brien, "Avalanche Breakdown Calculations for a Planar p-n Junction," IBM Journal 10, 213 (1966).

10. G. A. Baraff, "Distribution Functions and Ionization Rates for Hot Electrons in Semiconductors," Phys. Rev. 128, 2507, (1962).

11. C. R. Crowell and S. M. Sze, "Temperature Dependence of Avalanche Multiplication in Semiconductors," Appl. Phys. Letters, 9, 242 (1966).

12. R. A. Smith, SEMICONDUCTORS, p. 352, Cambridge Univ. Press, 1959.

13. S. S. Li, "The Dopant Density and Temperature Dependence of Hole Mobility and Resistivity in Boron-Doped Silicon, Solid State Electronics, 21, 1109 (1978).

14. S. M. Sze, Loc. cit., p. 27.

15. D. J. Hamilton and W. J. Howard, BASIC INTEGRATED CIRCUIT ENGINEERING, McGraw Hill, 1975, p. 192.

Appendix A

Computer Program Listing For Calculating Single and Two-Carrier Multiplication at Specified Voltage Increments

```
C       MAIN
C
C       EXECUTIVE PROGRAM
C
C       INITIALIZE ALL ARRAYS TO ZERO WITH A BLOCK DATA SUBPROGRAM
C
        BLOCK DATA
C
        COMMON/ARRAYS/ADUMMY(12493),IDUMMY(961)
        COMMON/COEFF/CDUMMY(4931)
        DATA ADUMMY/12493*0./,IDUMMY/961*0/
        DATA CDUMMY/4931*0.E0/
        END
C
C
C
        COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
       &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
       &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
        COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
       &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
        COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
       &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
        COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
        COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
        COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
        COMMON/SNGLE/VOLT0,DLTVLT,NVOLT,KOUTPT
        COMMON/SWTCH/ NDOPE
C
        DIMENSION TITLE (54)
        DIMENSION XPATH(100,3),YPATH(100,3),ET(100,3)
C
        NAMELIST/GEOM/XWIDTH,YDEPTH,IMAX,JMAX,NXWDTH,NYDPTH,
       &XDELTA,YDELTA,RAD
        NAMELIST/PRMTRS/ACCPTR,BRRIER,QSS,TMPTR
        NAMELIST/CONTRL/ITMAX,ITRMAX,CONVRG,SIDLT,AVDLT,VLTDLT,KNTMAX
        NAMELIST/SINGLE/VOLT0,DLTVLT,NVOLT,KOUTPT
        NAMELIST/SWITCH/ NDOPE
C
  100   READ(5,GEOM)
        READ(5,PRMTRS)
        READ(5,CONTRL)
        READ(5,SINGLE)
        READ(5,SWITCH)
        READ(5,1000) (TITLE(I),I=1,54)
 1000   FORMAT(18A4/18A4/18A4)
        IXWDTH=NXWDTH
        JYDPTH=JMAX-NYDPTH+1
        WRITE(6,1010)(TITLE(I),I=1,54)
 1010   FORMAT(1H1,60X,/,30X,18A4,/,30X,18A4,/,30X,18A4,///,5X,'SUMMARY
       &OF INPUT DATA :',//)
        WRITE(6,1020) XWIDTH,YDEPTH,IMAX,JMAX,NXWDTH,NYDPTH,
       &XDELTA,YDELTA,RAD
 1020   FORMAT(1H0,5X,'NAMELIST/GEOM/ XWIDTH=',1PE10.2,2X,'YDEPTH=',
       &1PE10.2,2X,'IMAX=',I3,2X,'JMAX=',I3,2X,'NXWDTH=',I3,2X,
       &'NYDPTH=',I3,2X,'XDELTA=',1PE10.2,/,6X,'YDELTA=',1PE10.2,2X,
       &'RAD=',1PE10.2,/)
```

```
      WRITE(6,1030)ACCPTR,BRRIER,QSS,TMPTR
 1030 FORMAT(1H0,5X,'NAMELIST/PRMTRS/ ACCPTR=',1PE10.2,2X,'BRRIER=',
     &1PE10.2,2X,'QSS=',1PE10.2,2X,'TMPTR=',1PE10.2)
      WRITE(6,1040) ITMAX,ITRMAX,CONVRG,SIDLT,AVDLT,VLTDLT,
     &KNTMAX
 1040 FORMAT(1H0,5X,'NAMELIST/CONTRL/ ITMAX=',I3,2X,'ITRMAX=',
     &I3,2X,'CONVRG=',1PE10.2,2X,'SIDLT=',1PE10.2,
     &/,6X,'AVDLT=',1PE10.2,2X,'VLTDLT=',1PE10.2,2X,'KNTMAX=',I4,/)
      WRITE(6,1050) VDLT0,DLTVLT,NVOLT,KOUTPT
 1050 FORMAT(1H0,5X,'NAMELIST/SINGLE/ VDLT0=',1PE10.2,2X,'DLTVLT=',
     &1PE10.2,2X,'NVOLT=',I2,2X,'KOUTPT=',I2,/)
      WRITE(6,1060) NOOPE
 1060 FORMAT(1H0,5X,'NAMELIST/SWITCH/ NOOPE=',I2,/)
C
      CALL AVLNCH(VLTDLT,AVDLT,XPATH,YPATH,ET,KNTMAX)
      GO TO 100
      END
C
C
```

```
      SUBROUTINE GRID
C
C     SETS UP THE GRADED LATTICE & COMPUTES THE FINITE-DIFFERENCE
C     COEFFICIENTS FOR PSI.
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
      COMMON/RSCALE/ RSFCTR
C
C     FIRST TREAT THE CORNER REGION BOUNDED BY XWIDTH & YDEPTH.
C
      IMX=IXWDTH-1
      X=XDELTA
      KEY=1
      GO TO 200
  100 XDLT(2)=XWIDTH*DLT
      DO 110 I=3,IXWDTH
  110 XDLT(I)=XDLT(I-1)*XDELTA
      IMX=JMAX-JYDPTH
      X=YDELTA
      KEY=2
      GO TO 200
  120 YDLT(JMAX)=YDEPTH*DLT
      DO 130 J=2,IMX
      JJ=JMAX+1-J
  130 YDLT(JJ)=YDLT(JJ+1)*YDELTA
C
C     NEXT TREAT THE REGION EXTERNAL TO THE CORNER REGION. THIS REGION
C     EXTENDS 1.25 DEPLETION WIDTHS TO THE RIGHT OF X=XWIDTH, AND
C     BELOW Y=YDEPTH. USE THE METHOD OF SECANTS TO DETERMINE THE
C     MULTIPLIERS ANALOGOUS TO XDELTA & YDELTA LEADING TO A SMOOTH
C     TRANSITION IN THE LATTICE SPACINGS AT THE INTERFACE OF THE
C     TWO REGIONS.
C
      KEY=3
      KED=1
      CC=XDLT(IXWDTH)/(1.25*RSFCTR)
      X1=1.
      IMX=IMAX-IXWDTH
      Y1=(1./IMX-CC)/CC
      GO TO 230
  140 XDLT(IMAX)=1.25*RSFCTR*DLT
      DO 150 I=2,IMX
      II=IMAX+1-I
  150 XDLT(II)=XDLT(II+1)*X
      KED=2
      CC=YDLT(JYDPTH+1)/(1.25*RSFCTR)
      X1=1.
```

```
      IMX=JYOPTH-1
      Y1=(1./IMX-CC)/CC
      GO TO 230
  160 YOLT(2)=1.25*RSFCTR*DLT
      DO 170 J=3,JYOPTH
  170 YOLT(J)=YOLT(J-1)*X
      XOLT(1)=XOLT(2)
      XOLT(IMAX+1)=XOLT(IMAX)
      YOLT(1)=YOLT(2)
      YOLT(JMAX+1)=YOLT(JMAX)
C
C     FINITE-DIFFERENCE COEFFICIENTS
C
      DO 175 I=1,IMAX
      DO 175 J=1,JMAX
      XAV=(XOLT(I)+XOLT(I+1))/2.
      YAV=(YOLT(J)+YOLT(J+1))/2.
      QAREA(I,J)=XAV*YAV
      HSTH(I,J)=XAV/YOLT(J)
      HWEST(I,J)=YAV/XOLT(I)
      HEAST(I,J)=YAV/XOLT(I+1)
      HNRTH(I,J)=XAV/YOLT(J+1)
  175 HCNTR(I,J)=-(HSTH(I,J)+HWEST(I,J)+HEAST(I,J)+HNRTH(I,J))
C
C     TABULATE THE X & Y COORDINATES OF THE LATTICE.
C
      XPOS(1)=0.
      YPOS(JMAX)=0.
      DO 180 I=2,IMAX
  180 XPOS(I)=XPOS(I-1)+XOLT(I)
      DO 190 J=2,JMAX
      JJ=JMAX+1-J
  190 YPOS(JJ)=YPOS(JJ+1)+YOLT(JJ+1)
      GO TO 270
C
C     INSTRUCTIONS 200 THROUGH 220 ARE A SUBPROGRAM FOR CALCULATING
C     THE WIDEST NORMALIZED INCREMENT DLT.
C
  200 EPSLON=1.-X
      IF(EPSLON.LT.1.E-3) GO TO 210
      DLT=(1.-X)/(1.-X**IMX)
      GO TO 220
  210 DLT=1.-(IMX-1)*EPSLON/2.
      DLT=1./(IMX*DLT)
  220 GO TO(100,120,250),KEY
C
C     INSTRUCTIONS 230 THROUGH 260 ARE A SUBPROGRAM FOR FINDING DLT
C     USING THE METHOD OF SECANTS.
C
C     FIRST INCREMENT X1 & X2 UNTIL Y1 & Y2 HAVE OPPOSITE SIGNS.
C
  230 X2=X1-0.02
      Y2=(1.-X2)/(1.-X2**IMX)
      Y2=Y2*X2**(IMX-1)
      Y2=(Y2-CC)/CC
      IF(Y1*Y2.LE.0.) GO TO 240
```

```
      X1=X2
      Y1=Y2
      GO TO 230
C
C     METHOD OF SECANTS
C
  240 DEN=Y2-Y1
      X=(X1*Y2-X2*Y1)/DEN
      GO TO 200
  250 Y=DLT*X**(IMX-1)
      Y=(Y-CC)/CC
      IF(ABS(Y).LT.0.01) GO TO 260
      X1=X2
      Y1=Y2
      X2=X
      Y2=Y
      GO TO 240
  260 GO TO(140,160),KED
  270 RETURN
      END
C
C
```

```
      SUBROUTINE BORDER
C
C     ASSIGNS BOUNDARY CONDITIONS BY MODIFYING THE FINITE-
C     DIFFERENCE COEFFICIENTS
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      CIRCLE(X,Y,R)=X*X+Y*Y-R*R
      POINT(Z,R)=SQRT(R*R-Z*Z)
C
      DO 90 I=1,IMAX
      DO 90 J=1,JMAX
      PSI(I,J)=0.
      HDENS(I,J)=0.
      ESUBX(I,J)=0.
      ESUBY(I,J)=0.
   90 IDENT(I,J)=0
C     THE LOWER BORDER, WHERE PSI=0
C
      DO 100 I=1,IMAX
      HSTH(I,1)=0.
      HWEST(I,1)=0.
      HCNTR(I,1)=1.E0
      HEAST(I,1)=0.
      HNRTH(I,1)=0.
  100 Q(I,1)=0.
C
C     THE RIGHT BORDER, WHERE PSI=0.
C
      DO 110 J=1,JMAX
      HSTH(IMAX,J)=0.
      HWEST(IMAX,J)=0.
      HCNTR(IMAX,J)=1.E0
      HEAST(IMAX,J)=0.
      HNRTH(IMAX,J)=0.
  110 Q(IMAX,J)=0.
C
C     THE UPPER BORDER, WHERE DPSI/DY=0.
C
      DO 120 I=IXWDTH,IMAX
      HNRTH(I,JMAX)=0.
  120 HSTH(I,JMAX)=2.E0*HSTH(I,JMAX)
C
C     THE LEFT BORDER, WHERE DPSI/DX=0.
C
```

```
      DO 130 J=1,JYOPTH
      HWEST(1,J)=0.
  130 HEAST(1,J)=2.E0*HEAST(1,J)
C
C     THE CORNER REGION
C
      X0=XPOS(IXWDTH)-RAD
      Y0=YPOS(JYOPTH)-RAD
      I=1
  140 J=JMAX+1
  150 J=J-1
      IF (J.LT.JYOPTH)GO TO 180
      IF (XPOS(I).GT.X0)GO TO 170
  160 HSTH(I,J)=0.
      HWEST(I,J)=0.
      HCNTR(I,J)=1.E0
      HEAST(I,J)=0.
      HNRTH(I,J)=0.
      Q(I,J)=1.0
      IDENT(I,J)=-1
      GO TO 150
  170 IF(YPOS(J).LE.Y0)GO TO 160
C
C     TEST TO DETERMINE IF THE POINT IS EXTERNAL TO THE CIRCLE.
C
      XP=XPOS(I)-X0
      YP=YPOS(J)-Y0
      TSTP=CIRCLE(XP,YP,RAD)
      IF(TSTP.LE.0.)GO TO 160
C
C     FIND THE WEST & NORTH DISTANCES TO THE BOUNDARY.  FIRST
C     DETERMINE WHICH POINTS ARE ALSO EXTERNAL, IF ANY.
C
      XW=XPOS(I-1)-X0
      YN=YPOS(J+1)-Y0
      TSTW=CIRCLE(XW,YP,RAD)
      TSTN=CIRCLE(XP,YN,RAD)
      XW=XPOS(I-1)
      IF(TSTW.LT.0.)XW=X0+POINT(YP,RAD)
      IF(TSTW.LT.0.) IDENT(I,J)=1
      XW=XPOS(I)-XW
      YN=YPOS(J+1)
      IF(TSTN.LT.0.)YN=Y0+POINT(XP,RAD)
      IF(TSTN.LT.0.) IDENT(I,J)=1
      YN=YPOS(J)-YN
      XWW=XDLT(I)/100.
      YNN=YDLT(J)/100.
      IF(XW.LT.XWW.OR.YN.LT.YNN) GO TO 160
C
C     MODIFY HWEST & HNRTH, TO ACCOUNT FOR THE CIRCULAR BOUNDARY.
C
      XAV=(XDLT(I+1)+XW)/2.E0
      YAV=(YN+YDLT(J))/2.E0
      HWEST(I,J)=YAV/XW
      HNRTH(I,J)=XAV/YN
      HCNTR(I,J)=-(HSTH(I,J)+HWEST(I,J)+HEAST(I,J)+HNRTH(I,J))
```

```
C
C       MODIFY QAREA TO ACCOUNT FOR THE PORTION (ASSUMED A
C       TRIANGLE) ECLIPSED BY THE CIRCLE.
C
        X1=XPOS(I)-XDLT(I)/2.E0
        Y1=YPOS(J)-YDLT(J+1)/2.E0
        XP=X1-X0
        YP=Y1-Y0
        TSTP=CIRCLE(XP,YP,RAD)
C
C       IF (TSTP.GE.0.) NO AREA IS ECLIPSED.
C
        IF(TSTP.GE.0.)GO TO 150
        Y2=POINT(XP,RAD)+Y0
        X3=POINT(YP,RAD)+X0
        QAREA(I,J)=QAREA(I,J)-(Y2-Y1)*(X3-X1)/2.E0
        GO TO 150
C
C       INDEX I & CONTINUE UNTIL I.GT.IXWDTH.
C
  180   I=I+1
        IF(I.LE.IXWDTH) GO TO 140
        IMAXX=IMAX-1
        DO 190 I=1,IMAXX
        DO 190 J=2,JMAX
  190   IF(IDENT(I,J).LT.0) PSI(I,J)=1.0
        RETURN
        END
C
C
```

```
      SUBROUTINE NITIAL
C
C     PROVIDES INITIAL ESTIMATE FOR PSI(I,J) USING CYLINDRICAL
C     COORDINATE DEPLETION THEORY.
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIOLT
      COMMON/NRMLZE/ONSNRM,PSINRM,DSTNRM,BLTZMN
C
C
C
      TRAD=RAD
      TX0=X0
      TY0=Y0
      IF(RAD.GT.0.) GO TO 90
      TRAD=0.05
      TX0=XWIDTH-TRAD
      TY0=YDEPTH-TRAD
   90 IF(RAD.EQ.XWIDTH.AND.RAD.EQ.YDEPTH) GO TO 190
C
C     THE HORIZONTAL SHEATH (TAPERED TO MATCH THE RADIAL SOLUTION AT
C     THE CIRCULAR CORNER).
C
      I=1
  100 RAD1=TRAD+(1.-TRAD)*(TX0-XPOS(I))/TX0
      Y00=YDEPTH-RAD1
      J=JYDPTH
      CALL DEPLTE(X,RAD1)
  110 RD=X*RAD1
  120 J=J-1
      RR=YPOS(J)-Y00
      IF(RR.GE.RD) GO TO 130
      PSI(I,J)=(RR-RD)*(RR+RD)/2.-RD*RD*ALOG(RR/RD)
      GO TO 120
  130 I=I+1
      IF(XPOS(I).GT.TX0) GO TO 140
      GO TO 100
C
C     THE VERTICAL SHEATH (TAPERED IN THE SAME MANNER).
C
  140 IM=I-1
      J=JMAX
  150 RAD1=TRAD+(1.-TRAD)*(TY0-YPOS(J))/TY0
      X00=XWIDTH-RAD1
      I=IXWDTH
      CALL DEPLTE(X,RAD1)
  160 RD=X*RAD1
  170 I=I+1
      RR=XPOS(I)-X00
```

```
      IF(RR.GE.RD) GO TO 180
      PSI(I,J)=(RR-RD)*(RR+RD)/2.-RD*RD*ALOG(RR/RD)
      GO TO 170
  180 J=J-1
      IF(YPOS(J).GT.TY0) GO TO 200
      GO TO 150
  190 I=0.
      J=JMAX
      GO TO 210
  200 I=IM
  210 RAD1=TRAD
C
C     THE CIRCULAR CORNER SHEATH
C
      CALL DEPLTE(X,RAD1)
  220 RD=X*RAD1
  230 I=I+1
      XP=XPOS(I)-TX0
      YP=YPOS(J)-TY0
      RR=SQRT(XP*XP+YP*YP)
      IF(RR.LE.RAD1) GO TO 230
      IF(RR.GE.RD) GO TO 240
      PSI(I,J)=(RR-RD)*(RR+RD)/2.-RD*RD*ALOG(RR/RD)
      GO TO 230
  240 J=J-1
      YP=YPOS(J)-TY0
      IF(YP.GE.RD) RETURN
      IF(RAD.EQ.XWIDTH.AND.RAD.EQ.YDEPTH) GO TO 250
      I=IM
      GO TO 230
  250 I=0
      GO TO 230
      END
```

```
      SUBROUTINE DEPLTE(X,RAD1)
C
C     FINDS THE DEPLETION RADIUS FROM A 1-D SOLUTION OF POISSON'S
C     EQUATION IN CYLINDRICAL COORDINATES.  RETURNS X=RD/RAD1.
C
      RHS(X)=1.-RAD1*RAD1*((1.-X*X)/2.+X*X*ALOG(X))
C
C     FOR RAD1 .GT.0.3, THE METHOD OF SECANTS IS USED TO FIND X.
C     FOR RAD1 .LT.0.3, A METHOD OF SUCCESSIVE APPROXIMATIONS WHICH
C     IS VERY FAST IS USED.  (THE LATTER FAILS TO CONVERGE WHEN RAD1
C     EXCEEDS 0.5.)
C
      IF(RAD1.GT.0.3) GO TO 110
      X1=(3.+4.*RAD1)/(4.*RAD1)
  100 X=SQRT((2./(RAD1*RAD1)-1.)/(2.*ALOG(X1)-1.))
      Y=RHS(X)
      IF(ABS(Y).LT.1.E-3) RETURN
      X1=X
      GO TO 100
C
C     METHOD OF SECANTS.  FIRST FIND X1 & X2 CAUSING Y1 & Y2 TO HAVE
C     OPPOSITE SIGNS.  THE STARTING POINT FOR X1 DIFFERS WHEN RAD1 .GT.
C     0.9, BASED ON EMPIRICAL KNOWLEDGE OF THE APPROXIMATE VALUE OF
C     THE ROOT TO SPEED UP THE SEARCH.
C
  110 IF(RAD1.GT.0.9) GO TO 120
      X1=4
      Y1=RHS(X1)
      GO TO 130
  120 X1=2
      Y1=RHS(X1)
  130 X2=X1-0.1
      Y2=RHS(X2)
      YY=Y1*Y2
      IF(YY.LT.0) GO TO 140
      X1=X2
      Y1=Y2
      GO TO 130
  140 X=(X1*Y2-X2*Y1)/(Y2-Y1)
      Y=RHS(X)
      IF(ABS(Y).LT.1.E-3) RETURN
      X1=X2
      Y1=Y2
      X2=X
      Y2=Y
      GO TO 140
      END
```

```
      SUBROUTINE GRELAX
C
C     IMPLEMENTS GUMMEL'S RELAXATION ALGORITHM FOR PSI
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      OMEGA=1.6
      IT=1
      IMAXX=IMAX-1
  100 DLTMX=0
      CALL DNSITY
      CALL POISSN
      DO 110 I=1,IMAXX
      DO 110 J=2,JMAX
      DLTMX=AMAX1(ABS(DELTA(I,J)),DLTMX)
      PSI(I,J)=PSI(I,J)+OMEGA*DELTA(I,J)
      IF(PSI(I,J).LT.0.) PSI(I,J)=0.
  110 CONTINUE
      DMX=0.
      DO 500 I=1,IMAXX
      DO 500 J=2,JMAX
      IF(ABS(DELTA(I,J)).LE.DMX) GO TO 500
      DMX=ABS(DELTA(I,J))
      II=I
      JJ=J
  500 CONTINUE
      WRITE(6,5000) DELTA(II,JJ),II,JJ
 5000 FORMAT(1H0,5X,'MAX DELTA IN GRELAX IS',1PE11.3,1X,
     &'AT COORDINATE (',I2,',',I2,')')
      IF(DLTMX.LE.CONVRG) GO TO 120
      IT=IT+1
      IF(IT.GT.ITMAX) GO TO 130
      GO TO 100
  120 WRITE(6,1000) IT
 1000 FORMAT(1H0,5X,I5,'  ITERATIONS WERE REQUIRED IN GRELAX',//)
      RETURN
  130 WRITE(6,1010) DLTMX,CONVRG
 1010 FORMAT(1H0,5X,'MAXIMUM ITERATIONS WERE EXCEEDED IN
     &GRELAX.    DLTMX=',1PE10.2,'  CONVRG=',1PE10.2,//)
      STOP
      END
C
C
```

```
      SUBROUTINE POISSN
C
C     SOLVES POISSONS EQUATION USING STONES METHOD.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,UBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      DIMENZR:TZRGEW(31,31)
C
      DO 100 I=1,IMAX
      DO 100 J=1,JMAX
      PSINEW(I,J)=PSI(I,J)
      B(I,J)=HSTH(I,J)
      D(I,J)=HWEST(I,J)
      E(I,J)=HCNTR(I,J)
      F(I,J)=HEAST(I,J)
  100 H(I,J)=HNRTH(I,J)
      CALL CHARGE
C
C     ITERATIVE STONE'S METHOD
C
      ISTONE=0
      ITR=1
  110 ISTONE=ISTONE+1
      CALL STONE(ISTONE,IMAX,JMAX,RESID,PSINEW)
      IF(RESID.LE.SIDLT) GO TO 120
      ITR=ITR+1
      IF(ITR.LE.ITRMAX) GO TO 110
      WRITE(6,1000)
 1000 FORMAT(1H0,1X,'MAXIMUM ITERATIONS EXCEEDED IN POISSN',/)
      STOP
  120 DO 130 I=1,IMAX
      DO 130 J=1,JMAX
  130 DELTA(I,J)=PSINEW(I,J)-PSI(I,J)
      WRITE(6,5000) ITR
 5000 FORMAT(2X,/,2X,I5,1X,'ITERATIONS WERE REQUIRED WITH STONES METHOD
     &IN POISSN')
      RETURN
      END
C
C
```

```
      SUBROUTINE DNSITY
C
C     USES BOLTZMANN DISTRIBUTIONS FOR THE HOLE & ELECTRON DENSITIES.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XOLT(32),YOLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIOLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      EFACTR=PSINRM/BLTZMN
      DO 130 I=1,IMAX
      DO 130 J=1,JMAX
      IF(IDENT(I,J).LT.0) GO TO 130
      EXPNT=PSI(I,J)*EFACTR
      IF(EXPNT.GT.80.) GO TO 100
      HDENS(I,J)=1.0/EXP(EXPNT)
      GO TO 110
  100 HDENS(I,J)=0.
  110 VOLT=PSINRM-VBLTIN
      EXPNT=2.*ALOG(TRNSIC/DNSNRM)+(PSINRM*PSI(I,J)-VOLT)/BLTZMN
      IF(EXPNT.LT.-80.) GO TO 120
      EDENS(I,J)=EXP(EXPNT)
      GO TO 130
  120 EDENS(I,J)=0.
  130 CONTINUE
      RETURN
      END
C
C
```

```
      SUBROUTINE CHARGE
C
C     CALCULATES THE CHARGE TERMS IN GUMMELS ALGORITHM
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      ALPHA=PSINRM/BLTZMN
      IMAXX=IMAX-1
      DO 100 I=1,IMAXX
      DO 100 J=2,JMAX
      IF(IDENT(I,J).LT.0) GO TO 100
      Q(I,J)=1.+EDENS(I,J)-HDENS(I,J)-ALPHA*(EDENS(I,J)+HDENS(I,J))*
     &PSI(I,J)
      Q(I,J)=2.*Q(I,J)*QAREA(I,J)
      E(I,J)=E(I,J)-2.*(EDENS(I,J)+HDENS(I,J))*ALPHA*QAREA(I,J)
  100 CONTINUE
C
C     MODIFY Q AT THE UPPER BORDER TO INCLUDE SURFACE CHARGE QSS.
C
      II=IXWDTH+1
      DO 110 I=II,IMAX
  110 Q(I,JMAX)=Q(I,JMAX)-QSS*(XDLT(I)+XDLT(I+1))
      RETURN
      END
C
C
```

```
      SUBROUTINE STONE(ISTONE,JMAX,KMAX,RESID,T)
C
C     IMPLEMENTS STONE'S METHOD FOR MATRIX FACTORIZATION.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      DIMENSION T(JMAX,KMAX)
      DIMENSION EE(31,31),FF(31,31),V(31,31),ALPHA(18)
C
C     DEFINE FREQUENTLY USED FUNCTIONS.(N FOR FORWARD K,R FOR REVERSE).
C
      BN(BDUMMY,EDUMMY)=BDUMMY/(1.E0+ALPH*EDUMMY)
      CN(DDUMMY,FDUMMY)=DDUMMY/(1.E0+ALPH*FDUMMY)
      DN1(EDUMMY,FDUMMY)=BB*(ALPH*EDUMMY-FDUMMY)
      DN2(FDUMMY,EDUMMY)=CC*(ALPH*FDUMMY-EDUMMY)
      EN(FDUMMY,EDUMMY)=(FDUMMY-ALPH*BB*EDUMMY)/DD
      FN(HDUMMY,FDUMMY)=(HDUMMY-ALPH*CC*FDUMMY)/DD
      BR(HDUMMY,EDUMMY)=HDUMMY/(1.E0+ALPH*EDUMMY)
      CR(DDUMMY,FDUMMY)=DDUMMY/(1.E0+ALPH*FDUMMY)
      DR1(EDUMMY,FDUMMY)=BB*(ALPH*EDUMMY-FDUMMY)
      DR2(FDUMMY,EDUMMY)=CC*(ALPH*FDUMMY-EDUMMY)
      ER(FDUMMY,EDUMMY)=(FDUMMY-ALPH*BB*EDUMMY)/DD
      FR(BDUMMY,FDUMMY)=(BDUMMY-ALPH*CC*FDUMMY)/DD
C
C     ISTONE FLAGS WHICH ALPHA TO USE AND WHETHER FORWARD (ISTONE ODD)
C     OR REVERSE (ISTONE EVEN) INDEXING OF K IS TO BE IMPLEMENTED.
C
      IF(ISTONE.GT.18) ISTONE=1
      ALPH=ALPHA(ISTONE)
      JMAXX=JMAX-1
      KMAXX=KMAX-1
      J=2*(ISTONE/2)
      IF(J.EQ.ISTONE) GO TO 170
C
C     ISTONE IS ODD.  INDEX K FORWARD
C
C
C     THE CORNER POINT (1,1)
C
      DD=E(1,1)
      EE(1,1)=F(1,1)/DD
      FF(1,1)=H(1,1)/DD
      R=Q(1,1)-(E(1,1)*T(1,1)+F(1,1)*T(2,1)+H(1,1)*T(1,2))
      RR=ABS(R)
      RESID=RR
      V(1,1)=R/DD
```

```
      IF(ABS(V(1,1)).LT.1.E-7) V(1,1)=0.
C
C     THE ROW K=1 - EXCLUDING THE RIGHT CORNER POINT.
C
      DO 100 J=2,JMAXX
      JP=J+1
      JN=J-1
      CC=CN(D(J,1),FF(JN,1))
      DD=E(J,1)+DN2(FF(JN,1),EE(JN,1))
      EE(J,1)=F(J,1)/DD
      FF(J,1)=FN(H(J,1),FF(JN,1))
      R=Q(J,1)-(D(J,1)*T(JN,1)+E(J,1)*T(J,1)+F(J,1)*T(JP,1)+H(J,1)*
     &T(J,2))
      RR=ABS(R)
      IF (RR.GT.RESID) RESID=RR
      V(J,1)=(R-CC*V(JN,1))/DD
  100 IF(ABS(V(J,1)).LT.1.E-7) V(J,1)=0.
C
C     THE CORNER POINT (JMAX,1)
C
      J=JMAX
      JN=JMAXX
      CC=CN(D(J,1),FF(JN,1))
      DD=E(J,1)+DN2(FF(JN,1),EE(JN,1))
      EE(J,1)=0.
      FF(J,1)=FN(H(J,1),FF(JN,1))
      R=Q(J,1)-(D(J,1)*T(JN,1)+E(J,1)*T(J,1)+H(J,1)*T(J,2))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(J,1)=(R-CC*V(JN,1))/DD
      IF(ABS(V(J,1)).LT.1.E-7) V(J,1)=0.
C
C     ALL ROWS EXCEPT ROW KMAX
C
      DO 120 K=2,KMAXX
      KP=K+1
      KN=K-1
C
C     THE LEFT BORDER POINT
C
      BB=BN(B(1,K),EE(1,KN))
      DD=E(1,K)+DN1(EE(1,KN),FF(1,KN))
      EE(1,K)=EN(F(1,K),EE(1,KN))
      FF(1,K)=H(1,K)/DD
      R=Q(1,K)-(B(1,K)*T(1,KN)+E(1,K)*T(1,K)+F(1,K)*T(2,K)+H(1,K)*
     &T(1,KP))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(1,K)=(R-BB*V(1,KN))/DD
      IF(ABS(V(1,K)).LT.1.E-7) V(1,K)=0.
C
C     THE ROW K EXCLUDING THE RIGHT BORDER POINT.
C
      DO 110 J=2,JMAXX
      JP=J+1
      JN=J-1
```

```
      BB=BN(B(J,K),EE(J,KN))
      CC=CN(D(J,K),FF(JN,K))
      DD=E(J,K)+DN1(EE(J,KN),FF(J,KN))+DN2(FF(JN,K),EE(JN,K))
      EE(J,K)=EN(F(J,K),EE(J,KN))
      FF(J,K)=FN(H(J,K),FF(JN,K))
      R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+
     &F(J,K)*T(JP,K)+H(J,K)*T(J,KP))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(J,K)=(R-BB*V(J,KN)-CC*V(JN,K))/DD
  110 IF(ABS(V(J,K)).LT.1.E-7) V(J,K)=0.
C
C     THE RIGHT BORDER POINT
C
      J=JMAX
      JN=JMAXX
      BB=BN(B(J,K),EE(J,KN))
      CC=CN(D(J,K),FF(JN,K))
      DD=E(J,K)+DN1(EE(J,KN),FF(J,KN))+DN2(FF(JN,K),EE(JN,K))
      EE(J,K)=0
      FF(J,K)=FN(H(J,K),FF(JN,K))
      R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+H(J,K)*
     &T(J,KP))
      RR=ABS(R)
      IF (RR.GT.RESID) RESID=RR
      V(J,K)=(R-BB*V(J,KN)-CC*V(JN,K))/DD
  120 IF(ABS(V(J,K)).LT.1.E-7) V(J,K)=0.
C
C     THE CORNER POINT (1,KMAX)
C
      K=KMAX
      KN=KMAXX
      BB=BN(B(1,K),EE(1,KN))
      DD=E(1,K)+DN1(EE(1,KN),FF(1,KN))
      EE(1,K)=EN(F(1,K),EE(1,KN))
      FF(1,K)=0
      R=Q(1,K)-(B(1,K)*T(1,KN)+E(1,K)*T(1,K)+F(1,K)*T(2,K))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(1,K)=(R-BB*V(1,KN))/DD
      IF(ABS(V(1,K)).LT.1.E-7) V(1,K)=0.
C
C     THE ROW KMAX, EXCLUDING THE CORNER POINT (JMAX,KMAX)
C
      DO 130 J=2,JMAXX
      JP=J+1
      JN=J-1
      BB=BN(B(J,K),EE(J,KN))
      CC=CN(D(J,K),FF(JN,K))
      DD=E(J,K)+DN1(EE(J,KN),FF(J,KN))+DN2(FF(JN,K),EE(JN,K))
      EE(J,K)=EN(F(J,K),EE(J,KN))
      FF(J,K)=0
      R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+F(J,K)*
     &T(JP,K))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
```

```
      V(J,K)=(R-BB*V(J,KN)-CC*V(JN,K))/DD
  130 IF(ABS(V(J,K)).LT.1.E-7) V(J,K)=0.
C
C     THE CORNER POINT (JMAX,KMAX)
C
      J=JMAX
      JN=JMAXX
      BB=BN(B(J,K),EE(J,KN))
      CC=CN(D(J,K),FF(JN,K))
      DD=E(J,K)+DN1(EE(J,KN),FF(J,KN))+DN2(FF(JN,K),EE(JN,K))
      EE(J,K)=0
      FF(J,K)=0
      R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(J,K)=(R-BB*V(J,KN)-CC*V(JN,K))/DD
      IF(ABS(V(J,K)).LT.1.E-7) V(J,K)=0.
C
C     INDEX K AND J IN REVERSE TO FIND DELTA
C
C     THE CORNER POINT (JMAX,KMAX)
C
      K=KMAX
      DELTA(JMAX,K)=V(JMAX,K)
C
C     THE ROW KMAX
C
      DO 140 JJ=1,JMAXX
      J=JMAXX+1-JJ
      JP=J+1
      DELTA(J,K)=V(J,K)-EE(J,K)*DELTA(JP,K)
  140 IF(ABS(DELTA(J,K)).LT.1.E-10) DELTA(J,K)=0.
C
C     THE ROW K, TREATING THE RIGHT BORDER FIRST.
C
      DO 150 KK=1,KMAXX
      K=KMAXX+1-KK
      KP=K+1
      DELTA(JMAX,K)=V(JMAX,K)-FF(JMAX,K)*DELTA(JMAX,KP)
      IF(ABS(DELTA(JMAX,K)).LT.1.E-10) DELTA(JMAX,K)=0.
      DO 150 JJ=1,JMAXX
      J=JMAXX+1-JJ
      JP=J+1
      DELTA(J,K)=V(J,K)-EE(J,K)*DELTA(JP,K)-FF(J,K)*DELTA(J,KP)
  150 IF(ABS(DELTA(J,K)).LT.1.E-10) DELTA(J,K)=0.
      GO TO 240
C
C     ISTONE IS EVEN.  INDEX K IN REVERSE ORDER.
C
C     THE CORNER POINT (1,KMAX)
C
  170 K=KMAX
      KN=KMAXX
      DD=E(1,K)
      EE(1,K)=F(1,K)/DD
      FF(1,K)=B(1,K)/DD
```

```
      R=Q(1,K)-(B(1,K)*T(1,KN)+E(1,K)*T(1,K)+F(1,K)*T(2,K))
      RR=ABS(R)
      RESID=RR
      V(1,K)=R/DD
      IF(ABS(V(1,K)).LT.1.E-7) V(1,K)=0.
C
C     THE ROW KMAX
C
      DO 180 J=2,JMAXX
      JP=J+1
      JN=J-1
      CC=CR(D(J,K),FF(JN,K))
      DD=E(J,K)+DR2(FF(JN,K),EE(JN,K))
      EE(J,K)=F(J,K)/DD
      FF(J,K)=FR(B(J,K),FF(JN,K))
      R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+F(J,K)*
     &T(JP,K))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(J,K)=(R-CC*V(JN,K))/DD
  180 IF(ABS(V(J,K)).LT.1.E-7) V(J,K)=0.
C
C     THE CORNER POINT (JMAX,KMAX)
C
      J=JMAX
      JN=JMAXX
      CC=CR(D(J,K),FF(JN,K))
      DD=E(J,K)+DR2(FF(JN,K),EE(JN,K))
      EE(J,K)=0
      FF(J,K)=FR(B(J,K),FF(JN,K))
      R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(J,K)=(R-CC*V(JN,K))/DD
      IF(ABS(V(J,K)).LT.1.E-7) V(J,K)=0.
C
C     ALL ROWS EXCEPT THE ROW K=1
C
      DO 200 KK=2,KMAXX
      K=KMAXX+2-KK
      KP=K+1
      KN=K-1
C
C     THE LEFT BORDER POINT (1,K)
C
      BB=BR(H(1,K),EE(1,KP))
      DD=E(1,K)+DR1(EE(1,KP),FF(1,KP))
      EE(1,K)=ER(F(1,K),EE(1,KP))
      FF(1,K)=B(1,K)/DD
      R=Q(1,K)-(B(1,K)*T(1,KN)+E(1,K)*T(1,K)+F(1,K)*T(2,K)+H(1,K)*
     &T(1,KP))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(1,K)=(R-BB*V(1,KP))/DD
      IF(ABS(V(1,K)).LT.1.E-7) V(1,K)=0.
C
```

```
C       THE ROW K EXCLUDING THE RIGHT BORDER POINT
C
        DO 190 J=2,JMAXX
        JP=J+1
        JN=J-1
        BB=BR(H(J,K),EE(J,KP))
        CC=CR(D(J,K),FF(JN,K))
        DD=E(J,K)+DR1(EE(J,KP),FF(J,KP))+DR2(FF(JN,K),EE(JN,K))
        EE(J,K)=ER(F(J,K),EE(J,KP))
        FF(J,K)=FR(B(J,K),FF(JN,K))
        R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+F(J,K)*
       &T(JP,K)+H(J,K)*T(J,KP))
        RR=ABS(R)
        IF(RR.GT.RESID) RESID=RR
        V(J,K)=(R-BB*V(J,KP)-CC*V(JN,K))/DD
  190   IF(ABS(V(J,K)).LT.1.E-7) V(J,K)=0.
C
C       THE RIGHT BORDER POINT
C
        J=JMAX
        JN=JMAXX
        BB=BR(H(J,K),EE(J,KP))
        CC=CR(D(J,K),FF(JN,K))
        DD=E(J,K)+DR1(EE(J,KP),FF(J,KP))+DR2(FF(JN,K),EE(JN,K))
        EE(J,K)=0.
        FF(J,K)=FR(B(J,K),FF(JN,K))
        R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+H(J,K)*
       &T(J,KP))
        RR=ABS(R)
        IF(RR.GT.RESID) RESID=RR
        V(J,K)=(R-BB*V(J,KP)-CC*V(JN,K))/DD
  200   IF(ABS(V(J,K)).LT.1.E-7) V(J,K)=0.
C
C       THE CORNER POINT (1,1)
C
        BB=BR(H(1,1),EE(1,2))
        DD=E(1,1)+DR1(EE(1,2),FF(1,2))
        EE(1,1)=ER(F(1,1),EE(1,2))
        FF(1,1)=0.
        R=Q(1,1)-(E(1,1)*T(1,1)+F(1,1)*T(2,1)+H(1,1)*T(1,2))
        RR=ABS(R)
        IF(RR.GT.RESID) RESID=RR
        V(1,1)=(R-BB*V(1,2))/DD
        IF(ABS(V(1,1)).LT.1.E-7) V(1,1)=0.
C
C       THE ROW K=1, EXCLUDING THE RIGHT CORNER POINT.
C
        DO 210 J=2,JMAXX
        JP=J+1
        JN=J-1
        BB=BR(H(J,1),EE(J,2))
        CC=CR(D(J,1),FF(JN,1))
        DD=E(J,1)+DR1(EE(J,2),FF(J,2))+DR2(FF(JN,1),EE(JN,1))
        EE(J,1)=ER(F(J,1),EE(J,2))
        FF(J,1)=0.
        R=Q(J,1)-(D(J,1)*T(JN,1)+E(J,1)*T(J,1)+F(J,1)*T(JP,1)+H(J,1)*
       &T(J,2))
```

```
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      U(J,1)=(R-BB*U(J,2)-CC*U(JN,1))/DD
  210 IF(ABS(U(J,1)).LT.1.E-7) U(J,1)=0.
C
C     THE CORNER POINT (JMAX,1)
C
      J=JMAX
      JN=JMAXX
      BB=BR(H(J,1),EE(J,2))
      CC=CR(D(J,1),FF(JN,1))
      DD=E(J,1)+DR1(EE(J,2),FF(J,2))+DR2(FF(JN,1),EE(JN,1))
      EE(J,1)=0.
      FF(J,1)=0.
      R=B(J,1)-(D(J,1)*T(JN,1)+E(J,1)*T(J,1)+H(J,1)*T(J,2))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      U(J,1)=(R-BB*U(J,2)-CC*U(JN,1))/DD
      IF(ABS(U(J,1)).LT.1.E-7) U(J,1)=0.
C
C     INDEX K FORWARD & J REVERSE TO FIND DELTA
C
C     THE POINT (JMAX,1)
C
      DELTA(JMAX,1)=U(JMAX,1)
C
C     THE ROW K=1
C
      DO 220 JJ=1,JMAXX
      J=JMAXX+1-JJ
      JP=J+1
      DELTA(J,1)=U(J,1)-EE(J,1)*DELTA(JP,1)
  220 IF(ABS(DELTA(J,1)).LT.1.E-10) DELTA(J,1)=0.
C
C     ALL REMAINING ROWS
C
      DO 230 K=2,KMAX
      KN=K-1
C
C     THE RIGHT BORDER POINT
C
      DELTA(JMAX,K)=U(JMAX,K)-FF(JMAX,K)*DELTA(JMAX,KN)
      IF(ABS(DELTA(JMAX,K)).LT.1.E-10) DELTA(JMAX,K)=0.
C
C     ALL REMAINING POINTS OF THE ROW
C
      DO 230 JJ=1,JMAXX
      J=JMAXX+1-JJ
      JP=J+1
      DELTA(J,K)=U(J,K)-EE(J,K)*DELTA(JP,K)-FF(J,K)*DELTA(J,KN)
  230 IF(ABS(DELTA(J,K)).LT.1.E-10) DELTA(J,K)=0.
C
C     CALCULATE NEW T(J,K)
C
  240 DO 250 J=1,JMAX
      DO 250 K=1,KMAX
```

```
160 RU=A**KHDT
170 I=I+1
    RR=XPOS(I)-X00
```



```
  250 T(J,K)=T(J,K)+DELTA(J,K)
      RETURN
C
C     ENTRY STONE1 SETS UP ALPHA VALUES & INITIALIZES ALL ARRAYS
C
      ENTRY STONE1(JMAX,KMAX)
C
      DO 320 J=2,JMAX
      DO 320 K=2,KMAX
      DX=XDLT(J)/XPOS(JMAX)
      DY=YDLT(K)/YPOS(1)
  320 EE(J,K)=2.E0/(1.E0/(DX*DX)+1.E0/(DY*DY))
C
C     FIND AVERAGE
C
      ALPHX=0
      DO 330 J=2,JMAX
      DO 330 K=2,KMAX
  330 ALPHX=ALPHX+EE(J,K)
      ALPHX=ALPHX/(JMAX*KMAX)
C
C     IMPLEMENT STONE'S EQ(27)
C
      DO 340 J=1,9
      ALPHY=J-1
      ALPHY=ALPHY/8.E0
  340 EE(J,1)=ALPHX**ALPHY
C
C     IMPLEMENT THE SEQUENCE OF 18 ALPHAS SUGGESTED BY STONE
C
      ALPHA(1)=EE(9,1)
      ALPHA(3)=EE(6,1)
      ALPHA(5)=EE(3,1)
      ALPHA(7)=EE(8,1)
      ALPHA(9)=EE(5,1)
      ALPHA(11)=EE(2,1)
      ALPHA(13)=EE(7,1)
      ALPHA(15)=EE(4,1)
      ALPHA(17)=EE(1,1)
      DO 350 J=2,18,2
  350 ALPHA(J)=ALPHA(J-1)
      DO 360 J=1,18
  360 ALPHA(J)=1.E0-ALPHA(J)
C
C
C     INITIALIZE ALL ARRAYS INTERNAL TO STONE TO ZERO.
C
      DO 370 J=1,JMAX
      DO 370 K=1,KMAX
      EE(J,K)=0.
      FF(J,K)=0.
      V(J,K)=0.
  370 DELTA(J,K)=0.
      RETURN
      END
C
```

```
      SUBROUTINE EFIELD
C
C     CALCULATES THE ELECTRIC FIELD COMPONENTS ESUBX & ESUBY
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
C     ESUBX COMPONENT
C
      IMAXX=IMAX-1
      DO 140 I=2,IMAXX
      DO 140 J=2,JMAX
      XW=XDLT(I)
      IF(IDENT(I,J).LT.0) GO TO 140
      IF(IDENT(I,J).EQ.0) GO TO 120
      IF(J.EQ.JMAX) GO TO 100
      YAV=(YDLT(J+1)+YDLT(J))/2.E0
      GO TO 110
  100 YAV=YDLT(J)
  110 XW=YAV/HWEST(I,J)
  120 ESUBX(I,J)=((PSI(I,J)-PSI(I+1,J))/XDLT(I+1)+(PSI(I-1,J)-PSI(I,J))
     &/XW)/2.E0
  140 CONTINUE
C
C     ESUBY COMPONENT
C
      JMAXX=JMAX-1
      DO 190 I=1,IMAXX
      DO 190 J=2,JMAXX
      YN=YDLT(J+1)
      IF(IDENT(I,J).LT.0) GO TO 190
      IF(IDENT(I,J).EQ.0) GO TO 170
      IF(I.EQ.1) GO TO 150
      XAV=(XDLT(I)+XDLT(I+1))/2.E0
      GO TO 160
  150 XAV=XDLT(2)
  160 YN=XAV/HNRTH(I,J)
  170 ESUBY(I,J)=((PSI(I,J+1)-PSI(I,J))/YN+(PSI(I,J)-PSI(I,J-1))/
     &YDLT(J))/2.E0
  190 CONTINUE
C
C     APPLY QUADRATIC EXTRAPOLATION TO FILL IN ESUBX & ESUBY AT THE
C     EXTERNAL POINTS BORDERING THE INTERFACE.
C
      DO 200 I=1,IMAX
      DO 200 J=1,JMAX
```

```
      IF(IDENT(I,J).GE.0) GO TO 200
      IF(IDENT(I+1,J).LT.0) GO TO 200
      DENOM=XDLT(I+3)*XDLT(I+2)*(XDLT(I+3)+XDLT(I+2))
      XX=XDLT(I+2)+XDLT(I+1)
      DENUM=XDLT(I+2)*ESUBX(I+3,J)+XDLT(I+3)*ESUBX(I+1,J)-
     &(XDLT(I+2)+XDLT(I+3))*ESUBX(I+2,J)
      AA=DENUM/DENOM
      DENUM=XDLT(I+2)*XDLT(I+2)*ESUBX(I+3,J)-XDLT(I+3)*XDLT(I+3)*
     &ESUBX(I+1,J)-(XDLT(I+2)*XDLT(I+2)-XDLT(I+3)*XDLT(I+3))*
     &ESUBX(I+2,J)
      BB=DENUM/DENOM
      ESUBX(I,J)=ESUBX(I+2,J)+AA*XX*XX-BB*XX
      DENOM=YDLT(J-2)*YDLT(J-1)*(YDLT(J-2)+YDLT(J-1))
      XX=YDLT(J-1)+YDLT(J)
      DENUM=YDLT(J-1)*ESUBY(I,J-3)+YDLT(J-2)*ESUBY(I,J-1)-
     &(YDLT(J-1)+YDLT(J-2))*ESUBY(I,J-2)
      AA=DENUM/DENOM
      DENUM=YDLT(J-1)*YDLT(J-1)*ESUBY(I,J-3)-YDLT(J-2)*YDLT(J-2)*
     &ESUBY(I,J-1)-(YDLT(J-1)*YDLT(J-1)-YDLT(J-2)*YDLT(J-2))*
     &ESUBY(I,J-2)
      BB=DENUM/DENOM
      ESUBY(I,J)=ESUBY(I,J-2)+AA*XX*XX-BB*XX
  200 CONTINUE
      RETURN
      END
C
C
```

```
C
C
      SUBROUTINE AVLNCH(VLTOLT,AVOLT,XPATH,YPATH,ET,KNTMAX)
C
C
C     EVALUATES THE BREAKDOWN VOLTAGE AFTER LOCATING THE TRAJECTORY
C     ALONG WHICH THE IONIZATION INTEGRAL IS A MAXIMUM.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XOLT(32),YOLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
      COMMON/SNGLE/VOLT0,DLTVLT,NVOLT,KOUTPT
      COMMON/RSCALE/ RSFCTR
C
      DIMENSION XSTART(3),YSTART(3),PI(3),KNTMX(3),ISTART(3),JSTART(3)
      DIMENSION XPATH(KNTMAX,3),YPATH(KNTMAX,3),ET(KNTMAX,3)
      DIMENSION RR(3),OMEG(3)
C
C     CALCULATE EQUILIBRIUM PARAMETERS. ESTIMATE THE BREAKDOWN VOLTAGE,
C     AND CALCULATE INITIAL NORMALIZATIONS.
C
      CALL EQULIB
      RSFCTR=1.0
      RDSAVE=RAD
      XWSAVE=XWIDTH
      YDSAVE=YDEPTH
      QSSAVE=QSS
      VOLT=VOLT0-DLTVLT
      N=0
   90 N=N+1
      IF(N.GT.NVOLT) RETURN
      VOLT=VOLT+DLTVLT
      PSINRM=VBLTIN+VOLT
      DSTNRM=SQRT(2.*11.7*8.854E-14*PSINRM/(1.602E-19*DNSNRM))
      IF(RDSAVE.NE.XWSAVE.OR.RDSAVE.NE.YDSAVE) GO TO 95
      RAD1=RDSAVE/(DSTNRM*1.E4)
      CALL DEPLTE(X,RAD1)
      RD=X*RAD1
      RSFCTR=RD-RAD1
   95 KSWCH=0
      K=1
      ISTART(2)=IXWDTH
      JSTART(2)=JYDPTH
C
C     PROVIDE CURRENT NORMALIZATIONS OF RAD,XWIDTH,YDEPTH, & QSS
C
  100 RAD=RDSAVE/(DSTNRM*1.E4)
```

```
      XWIDTH=XWSAVE/(DSTNRM*1.E4)
      YDEPTH=YDSAVE/(DSTNRM*1.E4)
      QSS=QSSAVE*DSTNRM/(11.7*8.854E-14*PSINRM)
      SINRM0=PSINRM
C
C     USING PRESENT NORMALIZATIONS, SOLVE POISSON'S EQ. & OBTAIN THE
C     ELECTRIC FIELD LATTICES. IF KSWCH.EQ.0, LOCATE POSITION OF MAX
C     E-FIELD AND ESTABLISH THREE STARTING LOCATIONS CENTERED ABOUT THAT
C     POSITION. IF KSWCH.GT.0, ESTABLISH ONLY THE CENTRAL STARTING
C     LOCATION.
C
      CALL GRID
      CALL STONE1(IMAX,JMAX)
      CALL BORDER
      CALL NITIAL
      CALL GRELAX
      CALL EFIELD
      I=ISTART(2)-1
      J=JSTART(2)-1
      IF(RAD.NE.XWIDTH.OR.RAD.NE.YDEPTH.OR.QSS.NE.0.) GO TO 105
      K=2
      KSWCH=1
      GO TO 120
  105 IF(KSWCH.GT.0) GO TO 120
      ETST=2.E5*DSTNRM/PSINRM
      EMAX=0.
      DO 110 II=1,IMAX
      DO 110 JJ=1,JMAX
      ETT=SQRT(ESUBX(II,JJ)*ESUBX(II,JJ)+ESUBY(II,JJ)*ESUBY(II,JJ))
      IF(ETT.LE.EMAX) GO TO 110
      EMAX=ETT
      I=II
      J=JJ
  110 CONTINUE
      IF(EMAX.GE.ETST) GO TO 115
      EMAX=EMAX*PSINRM/DSTNRM
      WRITE(6,1040) VOLT,EMAX
 1040 FORMAT(1H0,5X,'SKIPPING VOLT=',1PE10.2,' . MAX ELECTRIC FIELD
     &TOO SMALL (EMAX=',1PE10.2,')')
      GO TO 90
  115 I=I-2
      J=J-2
C
C     ESTABLISH STARTING COORDINATES FOR TRAJECTORY K.
C
  120 I=I+1
      J=J+1
      X=XPOS(I)
      Y=YPOS(J)
      IF(X.GT.X0) GO TO 125
      XSTART(K)=X
      YSTART(K)=YPOS(JYDPTH)
      GO TO 150
  125 IF(Y.GT.Y0) GO TO 130
      XSTART(K)=XPOS(IXWDTH)
      YSTART(K)=Y
```

```
      GO TO 150
  130 X=X-X0
      Y=Y-Y0
      IF(X.LT.Y) GO TO 140
      Y=Y/X
      XSTART(K)=X0+RAD/SQRT(1.E0+Y*Y)
      YSTART(K)=Y0+(XSTART(K)-X0)*Y
      GO TO 150
  140 X=X/Y
      YSTART(K)=Y0+RAD/SQRT(1.E0+X*X)
      XSTART(K)=X0+(YSTART(K)-Y0)*X
  150 IF(KSWCH.GT.0) GO TO 180
      K=K+1
      IF(K.GT.3) GO TO 170
      GO TO 120
C
C     TRACE OUT THE TRAJECTORY FOR STARTING POSITION K.
C
  170 K=1
  180 KNT=1
      KEXIT=0
      X=XSTART(K)
      Y=YSTART(K)
      I=IXWDTH
      J=JYDPTH
      DLTX=X-XPOS(IXWDTH)
      DLTY=Y-YPOS(JYDPTH)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      ISTART(K)=I
      JSTART(K)=J
      IF(DLTX.LT.XDLT(I+1)/1.E3) DLTX=XDLT(I+1)/1.E3
      IF(DLTY.LT.YDLT(J)/1.E3) DLTY=YDLT(J)/1.E3
      X=XPOS(I)+DLTX
      Y=YPOS(J)+DLTY
      CALL EFORCE(I,J,X,Y,DLTX,DLTY,EX,EY,ET(1,K))
      XPATH(1,K)=X
      YPATH(1,K)=Y
      ETT=ET(1,K)
  190 CALL TRAJEC(KNT,I,J,X,Y,DLTX,DLTY,ETT,KEXIT)
      IF(KEXIT.GT.0) GO TO 200
      KNT=KNT+1
      IF(KNT.GT.KNTMAX) GO TO 195
      XPATH(KNT,K)=X
      YPATH(KNT,K)=Y
      ET(KNT,K)=ETT
      GO TO 190
  195 WRITE(6,1000) KNTMAX
 1000 FORMAT(1H1,30X,/,5X,'EXCEEDED STORAGE ALLOTTED FOR TRAJECTORIES '
     &'(KNTMAX=',I3,').',/,5X,'INCREASE THE VALUE OF THAT PARAMETER ',
     &'IN NAMELIST CONTRL.',/,5X,'INCREASE DIMENSIONS ',
     &'ALLOTED TO XPATH,YPATH,& ET IN MAIN,IF NECESSARY.')
      STOP
  200 KNTMX(K)=KNT
C
C     APPLY PARABOLIC EXTRAPOLATION TO OBTAIN A BETTER ESTIMATE OF
C     ET(1,K) AT THE INTERFACE.
```

```
C
      DX=XPATH(4,K)-XPATH(3,K)
      DY=YPATH(4,K)-YPATH(3,K)
      DSP=SQRT(DX*DX+DY*DY)
      DX=XPATH(3,K)-XPATH(2,K)
      DY=YPATH(3,K)-YPATH(2,K)
      DSN=SQRT(DX*DX+DY*DY)
      DX=XPATH(3,K)-XPATH(1,K)
      DY=YPATH(3,K)-YPATH(1,K)
      DS=SQRT(DX*DX+DY*DY)
      DENOM=DSP*DSN*(DSP+DSN)
      DENUM=DSN*ET(4,K)+DSP*ET(2,K)-(DSP+DSN)*ET(3,K)
      AA=DENUM/DENOM
      DENUM=DSN*DSN*ET(4,K)-DSP*DSP*ET(2,K)-(DSN*DSN-DSP*DSP)*ET(3,K)
      BB=DENUM/DENOM
      ET(1,K)=ET(3,K)+AA*DS*DS-BB*DS
      KNT=1
      PI(K)=0.
      RR(K)=0.
      OMEG(K)=0.
      XPT=XPATH(1,K)*DSTNRM*1.E4
      YPT=YPATH(1,K)*DSTNRM*1.E4
      EPT=ET(1,K)*PSINRM/DSTNRM
      WRITE(6,1010) K,ISTART(K),JSTART(K),XPT,YPT,EPT,PI(K),OMEG(K)
 1010 FORMAT(2X,//,2X,'K',3X,'ISTART',1X,'JSTART',3X,'XPATH',5X,
     &'YPATH',7X,'ET',6X,'ALPHN',5X,'ALPHP',6X,'PI',7X,'OMEGA',
     &//,1X,I2,5X,I2,5X,I2,3X,1P3E10.2,20X,1P2E10.2)
C
C     EVALUATE THE IONIZATION INTEGRAL ALONG TRAJECTORY K.
C
  210 KNT=KNT+1
      IF(KNT.GT.KNTMX(K)) GO TO 220
      DX=XPATH(KNT,K)-XPATH(KNT-1,K)
      DY=YPATH(KNT,K)-YPATH(KNT-1,K)
      ET2=ET(KNT,K)
      ET1=ET(KNT-1,K)
      OMEG0=OMEG(K)
      CALL JNTGRL(DX,DY,ET1,ET2,PI(K),RR(K),OMEG(K),ALPHN,ALPHP)
      XPT=XPATH(KNT,K)*DSTNRM*1.E4
      YPT=YPATH(KNT,K)*DSTNRM*1.E4
      EPT=ET(KNT,K)*PSINRM/DSTNRM
      WRITE(6,1020) XPT,YPT,EPT,ALPHN,ALPHP,PI(K),OMEG(K)
 1020 FORMAT(20X,1P7E10.2)
      TST=(OMEG(K)-OMEG0)/OMEG(K)
      IF(TST.GT.1.E-4) GO TO 210
  220 IF(KSWCH.GT.0) GO TO 230
      K=K+1
      IF(K.LT.4) GO TO 180
      KSWCH=1
  230 IF(K.EQ.2) GO TO 290
C
C     ADJUST STARTING POSITIONS UNTIL THE TRAJECTORY K=2 PRODUCES THE
C     MAXIMUM IONIZATION INTEGRAL.
C
      IF(PI(1).GT.PI(2)) GO TO 250
      IF(K.EQ.1) GO TO 240
```

```
      IF(PI(3).GT.PI(2)) GO TO 270
  240 K=2
      GO TO 290
C
C     MAXIMUM IS TO LEFT OF K=2.  SHIFT LEFT ACCORDINGLY.
C
  250 K=1
      ISTART(2)=ISTART(1)
      JSTART(2)=JSTART(1)
      XSTART(2)=XSTART(1)
      YSTART(2)=YSTART(1)
      PI(2)=PI(1)
      KNTMX(2)=KNTMX(1)
      KMX=KNTMX(1)
      DO 260 KNT=1,KMX
      XPATH(KNT,2)=XPATH(KNT,1)
      YPATH(KNT,2)=YPATH(KNT,1)
  260 ET(KNT,2)=ET(KNT,1)
      I=ISTART(1)-2
      J=JSTART(1)-2
      IF(I.LT.0) GO TO 240
      GO TO 120
C
C     MAXIMUM IS TO RIGHT OF K=2.  SHIFT RIGHT ACCORDINGLY.
C
  270 K=3
      ISTART(2)=ISTART(3)
      JSTART(2)=JSTART(3)
      XSTART(2)=XSTART(3)
      YSTART(2)=YSTART(3)
      PI(2)=PI(3)
      KNTMX(2)=KNTMX(3)
      KMX=KNTMX(3)
      DO 280 KNT=1,KMX
      XPATH(KNT,2)=XPATH(KNT,3)
      YPATH(KNT,2)=YPATH(KNT,3)
  280 ET(KNT,2)=ET(KNT,3)
      I=ISTART(3)
      J=JSTART(3)
      IF(J.EQ.JMAX) GO TO 240
      GO TO 120
C
  290 EM1=EXP(PI(2))
      DEN=1.-OMEG(2)
      DN=ABS(DEN)
      IF(DN.LT.1.E-70) DEN=SIGN(1.E-70,DEN)
      EM2=1./DEN
      WRITE(6,1030) VOLT,EM1,EM2
 1030 FORMAT(1H0,5X,'VOLT=',1PE10.2,3X,'M1=',1PE10.2,3X,'M2=',1PE10.2)
      IF(KOUTPT.GT.0) GO TO 90
      CALL OUTPUT
      GO TO 90
      END
```

```
      SUBROUTINE VGUESS(VNIT)
C
C     SUPPLIES INITIAL ESTIMATE OF THE BREAKDOWN VOLTAGE USING AN
C     EMPIRICAL FIT TO THE KENNEDY-O'BRIEN CURVES.
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XOLT(32),YOLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIOLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      RR=RAD
      IF(RR.EQ.0) RR=0.1
      AA=1./(18.+150.*RR)
      VNIT=1./(AA+3.75E-12*DNSNRM**0.6)
      AA=2.303E4/(DNSNRM**.3695)
      BB=7.891E3/(DNSNRM**.2968)
      TT=TMPTR/300.-1.
      VNIT=VNIT*(AA*TT*TT+BB*TT+1.)
      RETURN
      END
C
C
```

```
      SUBROUTINE EQULIB
C
C     USING UNNORMALIZED VARIABLES, CALCULATES THE TEMPERATURE-
C     DEPENDENT BAND GAP, THE IONIZED ACCEPTOR DENSITY,AND THE BUILT-IN
C     POTENTIAL BARRIER, ACCOUNTING FOR THE TEMPERATURE DEPENDENCE OF
C     THE INTRINSIC DENSITY AND OF THE FERMI LEVEL.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
      COMMON/SWTCH/ NDOPE
C
      T=TMPTR/3.E2
      BLTZMN=0.02585*T
      EGAP=1.165-7.242E-3*T-3.664E-2*T*T
      EIV=EGAP/2.E0-1.306E-2*T
      X=EIV
      KEY=1
      GO TO 170
C
C     ADJUST X1&X2 UNTIL Y1&Y2 HAVE OPPOSITE SIGN
C
  100 X1=X
      Y1=Y
      DX=-EIV/2.E1
      IF(Y.LT.0.) DX=-DX
      X=X1+DX
      KEY=2
      GO TO 170
  110 X2=X
      Y2=Y
      IF(Y1*Y2.LE.0.) GO TO 140
  130 X1=X2
      Y1=Y2
      X=X2+DX
      GO TO 170
  140 KEY=3
C
C     METHOD OF SECANTS
C
  150 IF(ABS(Y).LT.1.E-3) GO TO 180
      DEN=Y2-Y1
      X=(X1*Y2-X2*Y1)/DEN
      GO TO 170
  160 X1=X2
      Y1=Y2
      X2=X
```

```
      Y2=Y
      GO TO 150
C
C     SUBROUTINE FOR EVALUATING Y
C
  170 EIF=X
      EFV=EIV-EIF
      EAF=4.38E-2-3.037E-8*ACCPTR**(1.0/3.0)-EFV
      BKGRND=ACCPTR/(1.0+(4.0+2.0/EXP(4.4E-2/BLTZMN))*EXP(EAF/BLTZMN))
      IF(NDOPE.GT.0) BKGRND=ACCPTR/(1.+2.*EXP(EAF/BLTZMN))
      TRNSIC=3.925E19*T*SQRT(T)*EXP(-EGAP/(2.*BLTZMN))
      BULKP=TRNSIC*EXP(EIF/BLTZMN)
      Y=(BULKP-BKGRND)/ACCPTR
      GO TO(100,110,160),KEY
C
C     CALCULATE THE BUILTIN POTENTIAL, THE ELECTRON DENSITY AT THE
C     METAL-SILICON INTERFACE, AND THE'BULK HOLE DENSITY.
C
  180 VBLTIN=BRRIER-EFV
      DNSNRM=BKGRND
      WRITE(6,1000)TMPTR,EGAP,EIV,EIF,EFV,TRNSIC,BKGRND,VBLTIN
 1000 FORMAT(2X,//,3X,'TMPTR',5X,'EGAP',7X,'EIV',7X,'EIF',7X,'EFV',
     &6X,'TRNSIC',4X,'BKGRND',4X,'VBLTIN',//,1X,1P8E10.2,//)
      RETURN
      END
C
C
```

```
      SUBROUTINE INDEX(I,J,DLTX,DLTY,KEXIT)
C
C     TRACKS WHICH 'BOX' OF THE LATTICE IS OCCUPIED BY THE MOVING
C     CHARGE & SUPPLIES ITS INDICES (I,J) AND CORRECT VALUES FOR
C     DLTX & DLTY WITH RESPECT TO THAT POINT. ALSO FLAGS AN EXIT OF
C     THE CHARGE FROM THE REGION.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      IF(DLTX.LT.0.) GO TO 100
      IF(DLTX.GT.XDLT(I+1)) GO TO 110
      GO TO 120
  100 IF(I.EQ.1) GO TO 150
      DLTX=XDLT(I)+DLTX
      I=I-1
      IF(DLTX.LT.0.) GO TO 100
      GO TO 120
  110 IF(I.EQ.(IMAX-1)) GO TO 150
      DLTX=DLTX-XDLT(I+1)
      I=I+1
      IF(DLTX.GT.XDLT(I+1)) GO TO 110
  120 IF(DLTY.LT.0.) GO TO 130
      IF(DLTY.GT.YDLT(J)) GO TO 140
      RETURN
  130 IF(J.EQ.JMAX) GO TO 150
      DLTY=YDLT(J+1)+DLTY
      J=J+1
      IF(DLTY.LT.0.) GO TO 130
      RETURN
  140 IF(J.EQ.2) GO TO 150
      DLTY=DLTY-YDLT(J)
      J=J-1
      IF(DLTY.GT.YDLT(J)) GO TO 140
      RETURN
  150 KEXIT=1
      RETURN
      END
C
C
```

```
      SUBROUTINE EFORCE(I,J,X,Y,DLTX,DLTY,EX,EY,ET)
C
C     APPLIES INTERPOLATION FOR THE X & Y COMPONENTS OF THE ELECTRIC
C     FIELD AT LOCATION (X,Y).
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
  100 EN=ESUBX(I,J)+DLTX*(ESUBX(I+1,J)-ESUBX(I,J))/XDLT(I+1)
      ES=ESUBX(I,J-1)+DLTX*(ESUBX(I+1,J-1)-ESUBX(I,J-1))/XDLT(I+1)
      EX=EN+DLTY*(ES-EN)/YDLT(J)
      EN=ESUBY(I,J)+DLTX*(ESUBY(I+1,J)-ESUBY(I,J))/XDLT(I+1)
      ES=ESUBY(I,J-1)+DLTX*(ESUBY(I+1,J-1)-ESUBY(I,J-1))/XDLT(I+1)
      EY=EN+DLTY*(ES-EN)/YDLT(J)
      ET=SQRT(EX*EX+EY*EY)
      RETURN
      END
C
C
```

```
      SUBROUTINE TRAJEC(KNT,I,J,X,Y,DLTX,DLTY,ET,KEXIT)
C
C     USES ADAMS-BASHFORTH-MOULTON PREDICTOR-CORRECTOR EQ'S TO TRACE
C     HOLE TRAJECTORIES, FOLLOWING A FOURTH-ORDER RUNGE-KUTTA STARTING
C     PROCEDURE.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,UBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      DIMENSION DX(7),DY(7)
C
      IF(KNT.GT.4) GO TO 100
      IF(KNT.GT.1) GO TO 90
      DLTA=0.005
      TOELT=DLTA/ET
      DO 80 K=1,7
      DX(K)=0.
   80 DY(K)=0.
C
C     RUNGE-KUTTA PROCEDURE
C
   90 CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      CALL EFORCE(I,J,X,Y,DLTX,DLTY,EX,EY,ET)
      DX1=EX*TOELT
      DY1=EY*TOELT
      XX=X+DX1/2.E0
      YY=Y+DY1/2.E0
      DLTX=XX-XPOS(I)
      DLTY=YY-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      CALL EFORCE(I,J,XX,YY,DLTX,DLTY,EX,EY,ET)
      DX2=EX*TOELT
      DY2=EY*TOELT
      XX=X+DX2/2.E0
      YY=Y+DY2/2.E0
      DLTX=XX-XPOS(I)
      DLTY=YY-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      CALL EFORCE(I,J,XX,YY,DLTX,DLTY,EX,EY,ET)
      DX3=EX*TOELT
      DY3=EY*TOELT
      XX=X+DX3
      YY=Y+DY3
      DLTX=XX-XPOS(I)
      DLTY=YY-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
```

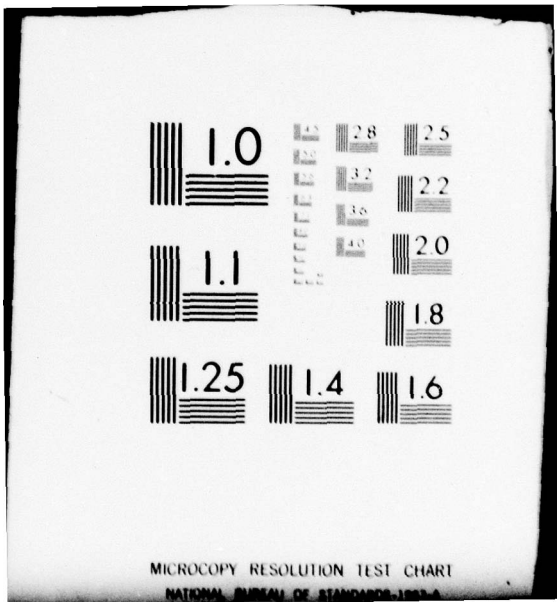
```
      CALL EFORCE(I,J,XX,YY,DLTX,DLTY,EX,EY,ET)
      DX4=EX*TDELT
      DY4=EY*TDELT
      DX(KNT+3)=(DX1+2.E0*(DX2+DX3)+DX4)/6.E0
      DY(KNT+3)=(DY1+2.E0*(DY2+DY3)+DY4)/6.E0
      X=X+DX(KNT+3)
      Y=Y+DY(KNT+3)
      DLTX=X-XPOS(I)
      DLTY=Y-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      CALL EFORCE(I,J,X,Y,DLTX,DLTY,EX,EY,ET)
      DX(KNT+3)=EX*TDELT
      DY(KNT+3)=EY*TDELT
      KOBLE=3
      RETURN
C
C     PREDICTOR-CORRECTOR, USING THE PRESENT INTERVAL
C
  100 XPROCT=(55.E0*DX(7)-59.E0*DX(6)+37.E0*DX(5)-9.E0*DX(4))/24.E0
      YPROCT=(55.E0*DY(7)-59.E0*DY(6)+37.E0*DY(5)-9.E0*DY(4))/24.E0
      XX=X+XPROCT
      YY=Y+YPROCT
      DLTX=XX-XPOS(I)
      DLTY=YY-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      IF(KEXIT.GT.0) RETURN
      CALL EFORCE(I,J,XX,YY,DLTX,DLTY,EX,EY,ET)
      DX1=EX*TDELT
      DY1=EY*TDELT
      XCRRCT=(9.E0*DX1+19.E0*DX(7)-5.E0*DX(6)+DX(5))/24.E0
      YCRRCT=(9.E0*DY1+19.E0*DY(7)-5.E0*DY(6)+DY(5))/24.E0
      DX2=ABS(XPROCT-XCRRCT)
      DY2=ABS(YPROCT-YCRRCT)
      DLT=AMAX1(DX2,DY2)
C
C     HALVE THE INTERVAL IF DLT EXCEEDS DLTA
C
      IF(DLT.LT.DLTA) GO TO 110
C
C     PARABOLIC INTERPOLATION COEFFICIENTS FOR THE HALF INTERVAL VALUES
C
      KOBLE=3
      TDELT=TDELT/2.E0
      AA=(DX(7)-2.E0*DX(6)+DX(5))/(8.E0*TDELT*TDELT)
      BB=(DX(7)-DX(5))/(4.E0*TDELT)
      AAA=(DX(6)-2.E0*DX(5)+DX(4))/(8.E0*TDELT*TDELT)
      BBB=(DX(6)-DX(4))/(4.E0*TDELT)
      DX(1)=DX(4)/2.E0
      DX(3)=DX(5)/2.E0
      DX(5)=DX(6)/2.E0
      DX(2)=(TDELT*(AAA*TDELT-BBB)+DX(3))/2.E0
      DX(4)=(TDELT*(AA*TDELT-BB)+DX(5))/2.E0
      DX(6)=(TDELT*(AA*TDELT+BB)+DX(5))/2.E0
      DX(7)=DX(7)/2.E0
      AA=(DY(7)-2.E0*DY(6)+DY(5))/(8.E0*TDELT*TDELT)
      BB=(DY(7)-DY(5))/(4.E0*TDELT)
```

```
      AAA=(DY(6)-2.E0*DY(5)+DY(4))/(8.E0*TDELT*TDELT)
      BBB=(DY(6)-DY(4))/(4.E0*TDELT)
      DY(1)=DY(4)/2.E0
      DY(3)=DY(5)/2.E0
      DY(5)=DY(6)/2.E0
      DY(2)=(TDELT*(AAA*TDELT-BBB)+DY(3))/2.E0
      DY(4)=(TDELT*(AA*TDELT-BB)+DY(5))/2.E0
      DY(6)=(TDELT*(AA*TDELT+BB)+DY(5))/2.E0
      DY(7)=DY(7)/2.E0
      GO TO 100
C
C     TENTATIVELY DOUBLE THE INTERVAL,BUT ONLY IF 3 STEPS HAVE BEEN
C     TAKEN SINCE THE LAST DOUBLING OR HALVING, AND ONLY IF DLT
C     IS LESS THAN ONE HALF OF DLTA
C
  110 IF(KDBLE.GT.0) GO TO 120
      DLT=DLT*2.0
      IF(DLT.GT.DLTA) GO TO 120
      XPRDT=(55.E0*DX(7)-59.E0*DX(5)+37.E0*DX(3)-9.E0*DX(1))/12.E0
      YPRDT=(55.E0*DY(7)-59.E0*DY(5)+37.E0*DY(3)-9.E0*DY(1))/12.E0
      XX=X+XPRDT
      YY=Y+YPRDT
      DLTX=XX-XPOS(I)
      DLTY=YY-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      IF(KEXIT.GT.0) GO TO 120
      CALL EFORCE(I,J,XX,YY,DLTX,DLTY,EX,EY,ET)
      DX1=EX*TDELT
      DY1=EY*TDELT
      XCRRT=(9.E0*DX1+19.E0*DX(7)-5.E0*DX(5)+DX(3))/12.E0
      YCRRT=(9.E0*DY1+19.E0*DY(7)-5.E0*DY(5)+DY(3))/12.E0
      DX2=ABS(XPRDT-XCRRT)
      DY2=ABS(YPRDT-YCRRT)
      DLT=AMAX1(DX2,DY2)
      IF(DLT.GT.DLTA) GO TO 120
C
C     DOUBLE INTERVAL IS OK, SO DO IT
C
      DX(7)=DX(7)*2.E0
      DX(6)=DX(5)*2.E0
      DX(5)=DX(3)*2.E0
      DX(4)=DX(1)*2.E0
      DY(7)=DY(7)*2.E0
      DY(6)=DY(5)*2.E0
      DY(5)=DY(3)*2.E0
      DY(4)=DY(1)*2.E0
      X=X+(19.E0*XPRDT+251.E0*XCRRT)/270.E0
      Y=Y+(19.E0*YPRDT+251.E0*YCRRT)/270.E0
      TDELT=2.E0*TDELT
      KDBLE=3
      GO TO 130
C
C     NO DOUBLING OCCURRED. COMPUTE NEW X & Y. SHIFT INCREMENTS LEFT
C     & UPDATE THE MOST RECENT INCREMENT.
C
  120 X=X+(19.E0*XPRDCT+251.E0*XCRRCT)/270.E0
```

AD-A075 010 APPLIED ELECTRONIC RESEARCH CORP GAINESVILLE FL F/G 20/12
TWO-DIMENSIONAL AVALANCHE BREAKDOWN IN PT-SI SCHOTTKY BARRIER D--ETC(U)
JUL 79 D P KENNEDY F19628-77-C-0220
UNCLASSIFIED RADC-TR-79-189 NL

2 OF 2
AD A075010

END
DATE FILMED
11-79
DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
      Y=Y+(19.E0*YPRDCT+251.E0*YCRRCT)/270.E0
      KDBLE=KDBLE-1
  130 DO 140 K=1,6
      DX(K)=DX(K+1)
  140 DY(K)=DY(K+1)
      DLTX=X-XPOS(I)
      DLTY=Y-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      IF(KEXIT.GT.0) RETURN
      CALL EFORCE(I,J,X,Y,DLTX,DLTY,EX,EY,ET)
      ETT=ET*PSINRM/DSTNRM
      IF(ETT.LT.2.E4) KEXIT=1
      DX(7)=EX*TDELT
      DY(7)=EY*TDELT
      RETURN
      END
C
C
```

```
      SUBROUTINE JNTGRL(DX,DY,ET1,ET2,PI,RR,OMEG,ALPHN,ALPHP)
C
C     USES TRAPEZOIDAL RULE FOR THE TWO-CARRIER MULTIPLICATION INTEGRAL.
C     (USES DIMENSIONED VARIABLES.)
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      DS=SQRT(DX*DX+DY*DY)*DSTNRM
      ET=(ET1+(ET2-ET1)/4.)*PSINRM/DSTNRM
      CALL TWNSND(ET,ALPHN,ALPHP)
      RR=RR+(ALPHN-ALPHP)*DS/2.
      ET=((ET1+ET2)/2.)*PSINRM/DSTNRM
      CALL TWNSND(ET,ALPHN,ALPHP)
      OMEG=OMEG+ALPHP*EXP(RR)*DS
      PI=PI+ALPHN*DS
      ET=(ET1+3.*(ET2-ET1)/4.)*PSINRM/DSTNRM
      CALL TWNSND(ET,ALPHN,ALPHP)
      RR=RR+(ALPHN-ALPHP)*DS/2.
      RETURN
      END
```

```
      SUBROUTINE TWNSND(ET,ALPHN,ALPHP)
C
C     CALCULATES THE TOWNSEND IONIZATION COEFFICIENTS FOR ELECTRONS AND
C     HOLES USING CROWELL & SZE'S EMPIRICAL FIT FOR BARAFF'S CURVES.
C     USES UNNORMALIZED VARIABLES. (SEE S.M.SZE, PHYSICS OF
C     SEMICONDUCTOR DEVICES, PP 61-65.)
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,3
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      EXPNT(R2,R1,X2,X1)=(11.5*R2-1.17*R1+3.9E-4)*X2 +
     &(46.*R2-11.9*R1+1.75E-2)*X1-(757.*R2-75.5*R1+1.91)
      TANH(X)=(EXP(X)-EXP(-X))/(EXP(X)+EXP(-X))
      XX=3.15E-2/BLTZMN
      EP=6.3E-2*TANH(XX)
      ELMBDA=76.0E-8*EP/6.3E-2
      EI=1.5*EGAP
      R1=EP/EI
      R2=R1*R1
      X1=EI/(ET*ELMBDA)
      X2=X1*X1
      XN=EXPNT(R2,R1,X2,X1)
      IF(XN.LT.-80.) XN=-80.
      ALPHN=EXP(XN)/ELMBDA
      ELMBDA=50.0E-8*EP/6.3E-2
      X1=EI/(ET*ELMBDA)
      X2=X1*X1
      XN=EXPNT(R2,R1,X2,X1)
      IF(XN.LT.-80.) XN=-80.
      ALPHP=EXP(XN)/ELMBDA
      RETURN
      END
```

```
      SUBROUTINE OUTPUT
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIOLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      DIMENSION SCRTCH(31),X(31),Y(31)
C
      DO 80 I=1,IMAX
   80 X(I)=XPOS(I)*DSTNRM*1.0E4
      DO 90 J=1,JMAX
   90 Y(J)=YPOS(J)*DSTNRM*1.0E4
      WRITE(6,1000)
 1000 FORMAT(1H1,60X,/,32X,'X(I),I=1,IMAX',//)
      WRITE(6,1010)(X(I),I=1,IMAX)
 1010 FORMAT(32X,1P10E10.2)
      WRITE(6,1020)
 1020 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'PSI(I,J),I=1,IMAX',//)
      DO 110 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 100 I=1,IMAX
  100 SCRTCH(I)=PSI(I,J)*PSINRM
      WRITE(6,1030) J,Y(J),(SCRTCH(I),I=1,10)
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
 1030 FORMAT(1X,' ',/,I7,5X,1PE10.2,10X,1P10E10.2)
  110 CONTINUE
 1040 FORMAT(32X,1P10E10.2)
      WRITE(6,1000)
      WRITE(6,1010)(X(I),I=1,IMAX)
      WRITE(6,1060)
 1060 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'HDENS(I,J),I=1,IMAX',//)
      DO 150 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 140 I=1,IMAX
  140 SCRTCH(I)=HDENS(I,J)*DNSNRM
      WRITE(6,1030) J,Y(J),(SCRTCH(I),I=1,10)
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
  150 CONTINUE
      WRITE(6,1000)
      WRITE(6,1010)(X(I),I=1,IMAX)
      WRITE(6,1100)
 1100 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'EDENS(I,J),I=1,IMAX',//)
      DO 154 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 152 I=1,IMAX
  152 SCRTCH(I)=EDENS(I,J)*DNSNRM
      WRITE(6,1030)J,Y(J),(SCRTCH(I),I=1,10)
```

```
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
  154 CONTINUE
      WRITE(6,1000)
      WRITE(6,1010)(X(I),I=1,IMAX)
      WRITE(6,1070)
 1070 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'ESUBX(I,J),I=1,IMAX',//)
      DO 170 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 160 I=1,IMAX
  160 SCRTCH(I)=ESUBX(I,J)*PSINRM/DSTNRM
      WRITE(6,1030) J,Y(J),(SCRTCH(I),I=1,10)
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
  170 CONTINUE
      WRITE(6,1000)
      WRITE(6,1010)(X(I),I=1,IMAX)
      WRITE(6,1080)
 1080 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'ESUBY(I,J),I=1,IMAX',//)
      DO 190 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 180 I=1,IMAX
  180 SCRTCH(I)=ESUBY(I,J)*PSINRM/DSTNRM
      WRITE(6,1030) J,Y(J),(SCRTCH(I),I=1,10)
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
  190 CONTINUE
      WRITE(6,1000)
      WRITE(6,1010)(X(I),I=1,IMAX)
      WRITE(6,1090)
 1090 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'ETOTAL(I,J),I=1,IMAX',//)
      DO 210 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 200 I=1,IMAX
  200 SCRTCH(I)=SQRT(ESUBX(I,J)*ESUBX(I,J)+ESUBY(I,J)*ESUBY(I,J))*
     &PSINRM/DSTNRM
      WRITE(6,1030) J,Y(J),(SCRTCH(I),I=1,10)
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
  210 CONTINUE
      RETURN
      END
```

Appendix B

Computer Program Listing For Calculation Avalanche Breakdown in a Pt-Si Schottky Barrier Diode

```
C       MAIN
C
C       EXECUTIVE PROGRAM
C
C       INITIALIZE ALL ARRAYS TO ZERO WITH A BLOCK DATA SUBPROGRAM
C
        BLOCK DATA
C
        COMMON/ARRAYS/ADUMMY(12493),IDUMMY(961)
        COMMON/COEFF/COUMMY(4931)
        DATA ADUMMY/12493*0./,IDUMMY/961*0/
        DATA COUMMY/4931*0.E0/
        END
C
C
C
        COMMON/ARRAYS/PSI(31,31),EQENS(31,31),HQENS(31,31),B(31,31),
       &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
       &QELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
        COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
       &HNRTH(31,31),XQLT(32),YQLT(32),XPOS(31),YPOS(31)
        COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,UBLTIN,VQLT,
       &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
        COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
        COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIQLT
        COMMON/NRMLZE/QNSNRM,PSINRM,QSTNRM,BLTZMN
        COMMON/SWTCH/NQOPE
C
        DIMENSION TITLE (54)
        DIMENSION XPATH(100,3),YPATH(100,3),ET(100,3)
C
        NAMELIST/GEOM/XWIDTH,YDEPTH,IMAX,JMAX,NXWDTH,NYDPTH,
       &XDELTA,YDELTA,RAD
        NAMELIST/PRMTRS/ACCPTR,BRRIER,QSS,TMPTR
        NAMELIST/CONTRL/ITMAX,ITRMAX,CONVRG,SIQLT,AVQLT,VLTQLT,KNTMAX
        NAMELIST/SWITCH/NQOPE
C
  100   READ(5,GEOM)
        READ(5,PRMTRS)
        READ(5,CONTRL)
        READ(5,SWITCH)
        READ(5,1000) (TITLE(I),I=1,54)
 1000   FORMAT(18A4/18A4/18A4)
        IXWDTH=NXWDTH
        JYDPTH=JMAX-NYDPTH+1
        WRITE(6,1010)(TITLE(I),I=1,54)
 1010   FORMAT(1H1,60X,/,30X,18A4,/,30X,18A4,/,30X,18A4,///,5X,'SUMMARY
       &OF INPUT DATA :',//)
        WRITE(6,1020) XWIDTH,YDEPTH,IMAX,JMAX,NXWDTH,NYDPTH,
       &XDELTA,YDELTA,RAD
 1020   FORMAT(1H0,5X,'NAMELIST/GEOM/ XWIDTH=',1PE10.2,2X,'YDEPTH=',
       &1PE10.2,2X,'IMAX=',I3,2X,'JMAX=',I3,2X,'NXWDTH=',I3,2X,
       &'NYDPTH=',I3,2X,'XDELTA=',1PE10.2,/,6X,'YDELTA=',1PE10.2,2X,
       &'RAD=',1PE10.2,/)
        WRITE(6,1030)ACCPTR,BRRIER,QSS,TMPTR
 1030   FORMAT(1H0,5X,'NAMELIST/PRMTRS/ ACCPTR=',1PE10.2,2X,'BRRIER=',
       &1PE10.2,2X,'QSS=',1PE10.2,2X,'TMPTR=',1PE10.2)
```

```
      WRITE(6,1040) ITMAX,ITRMAX,CONVRG,SIDLT,AVDLT,VLTDLT,
     &KNTMAX
 1040 FORMAT(1H0,5X,'NAMELIST/CONTRL/ ITMAX=',I3,2X,'ITRMAX=',
     &I3,2X,'CONVRG=',1PE10.2,2X,'SIDLT=',1PE10.2,
     &/,6X,'AVDLT=',1PE10.2,2X,'VLTDLT=',1PE10.2,2X,'KNTMAX=',I4,/)
      WRITE (6,1060) NDOPE
 1060 FORMAT(1H0,5X,'NAMELIST/SWITCH/ NDOPE=',I2,/)
C
      CALL INTGR1
      CALL AVLNCH(VLTDLT,AVDLT,XPATH,YPATH,ET,KNTMAX)
      CALL OUTPUT
      VOLT=PSINRM-VBLTIN
      WRITE(6,1050) VOLT
 1050 FORMAT(1H0,5X,'BREAKDOWN VOLTAGE=',1PE10.2)
      GO TO 100
      END
C
C
```

```
      SUBROUTINE GRID
C
C     SETS UP THE GRADED LATTICE & COMPUTES THE FINITE-DIFFERENCE
C     COEFFICIENTS FOR PSI.
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
      COMMON/RSCALE/ RSFCTR
C
C     FIRST TREAT THE CORNER REGION BOUNDED BY XWIDTH & YDEPTH.
C
      IMX=IXWDTH-1
      X=XDELTA
      KEY=1
      GO TO 200
  100 XDLT(2)=XWIDTH*DLT
      DO 110 I=3,IXWDTH
  110 XDLT(I)=XDLT(I-1)*XDELTA
      IMX=JMAX-JYDPTH
      X=YDELTA
      KEY=2
      GO TO 200
  120 YDLT(JMAX)=YDEPTH*DLT
      DO 130 J=2,IMX
      JJ=JMAX+1-J
  130 YDLT(JJ)=YDLT(JJ+1)*YDELTA
C
C     NEXT TREAT THE REGION EXTERNAL TO THE CORNER REGION. THIS REGION
C     EXTENDS 1.25 DEPLETION WIDTHS TO THE RIGHT OF X=XWIDTH, AND
C     BELOW Y=YDEPTH. USE THE METHOD OF SECANTS TO DETERMINE THE
C     MULTIPLIERS ANALOGOUS TO XDELTA & YDELTA LEADING TO A SMOOTH
C     TRANSITION IN THE LATTICE SPACINGS AT THE INTERFACE OF THE
C     TWO REGIONS.
C
      KEY=3
      KED=1
      CC=XDLT(IXWDTH)/(1.25*RSFCTR)
      X1=1.
      IMX=IMAX-IXWDTH
      Y1=(1./IMX-CC)/CC
      GO TO 230
  140 XDLT(IMAX)=1.25*RSFCTR*DLT
      DO 150 I=2,IMX
      II=IMAX+1-I
  150 XDLT(II)=XDLT(II+1)*X
      KED=2
      CC=YDLT(JYDPTH+1)/(1.25*RSFCTR)
      X1=1.
```

```
      IMX=JYDPTH-1
      Y1=(1./IMX-CC)/CC
      GO TO 230
  160 YDLT(2)=1.25*RSFCTR*DLT
      DO 170 J=3,JYDPTH
  170 YDLT(J)=YDLT(J-1)*X
      XDLT(1)=XDLT(2)
      XDLT(IMAX+1)=XDLT(IMAX)
      YDLT(1)=YDLT(2)
      YDLT(JMAX+1)=YDLT(JMAX)
C
C     FINITE-DIFFERENCE COEFFICIENTS
C
      DO 175 I=1,IMAX
      DO 175 J=1,JMAX
      XAV=(XDLT(I)+XDLT(I+1))/2.
      YAV=(YDLT(J)+YDLT(J+1))/2.
      QAREA(I,J)=XAV*YAV
      HSTH(I,J)=XAV/YDLT(J)
      HWEST(I,J)=YAV/XDLT(I)
      HEAST(I,J)=YAV/XDLT(I+1)
      HNRTH(I,J)=XAV/YDLT(J+1)
  175 HCNTR(I,J)=-(HSTH(I,J)+HWEST(I,J)+HEAST(I,J)+HNRTH(I,J))
C
C     TABULATE THE X & Y COORDINATES OF THE LATTICE.
C
      XPOS(1)=0.
      YPOS(JMAX)=0.
      DO 180 I=2,IMAX
  180 XPOS(I)=XPOS(I-1)+XDLT(I)
      DO 190 J=2,JMAX
      JJ=JMAX+1-J
  190 YPOS(JJ)=YPOS(JJ+1)+YDLT(JJ+1)
      GO TO 270
C
C     INSTRUCTIONS 200 THROUGH 220 ARE A SUBPROGRAM FOR CALCULATING
C     THE WIDEST NORMALIZED INCREMENT DLT.
C
  200 EPSLON=1.-X
      IF(EPSLON.LT.1.E-3) GO TO 210
      DLT=(1.-X)/(1.-X**IMX)
      GO TO 220
  210 DLT=1.-(IMX-1)*EPSLON/2.
      DLT=1./(IMX*DLT)
  220 GO TO(100,120,250),KEY
C
C     INSTRUCTIONS 230 THROUGH 260 ARE A SUBPROGRAM FOR FINDING DLT
C     USING THE METHOD OF SECANTS.
C
C     FIRST INCREMENT X1 & X2 UNTIL Y1 & Y2 HAVE OPPOSITE SIGNS.
C
  230 X2=X1-0.02
      Y2=(1.-X2)/(1.-X2**IMX)
      Y2=Y2*X2**(IMX-1)
      Y2=(Y2-CC)/CC
      IF(Y1*Y2.LE.0.) GO TO 240
```

```
      X1=X2
      Y1=Y2
      GO TO 230
C
C     METHOD OF SECANTS
C
  240 DEN=Y2-Y1
      X=(X1*Y2-X2*Y1)/DEN
      GO TO 200
  250 Y=DLT*X**(IMX-1)
      Y=(Y-CC)/CC
      IF(ABS(Y).LT.0.01) GO TO 260
      X1=X2
      Y1=Y2
      X2=X
      Y2=Y
      GO TO 240
  260 GO TO(140,160),KEO
  270 RETURN
      END
C
C
```

```
      SUBROUTINE BORDER
C
C     ASSIGNS BOUNDARY CONDITIONS BY MODIFYING THE FINITE-
C     DIFFERENCE COEFFICIENTS
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIOLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      CIRCLE(X,Y,R)=X*X+Y*Y-R*R
      POINT(Z,R)=SQRT(R*R-Z*Z)
C
      DO 90 I=1,IMAX
      DO 90 J=1,JMAX
      PSI(I,J)=0.
      HDENS(I,J)=0.
      ESUBX(I,J)=0.
      ESUBY(I,J)=0.
   90 IDENT(I,J)=0
C     THE LOWER BORDER, WHERE PSI=0
C
      DO 100 I=1,IMAX
      HSTH(I,1)=0.
      HWEST(I,1)=0.
      HCNTR(I,1)=1.E0
      HEAST(I,1)=0.
      HNRTH(I,1)=0.
  100 Q(I,1)=0.
C
C     THE RIGHT BORDER, WHERE PSI=0.
C
      DO 110 J=1,JMAX
      HSTH(IMAX,J)=0.
      HWEST(IMAX,J)=0.
      HCNTR(IMAX,J)=1.E0
      HEAST(IMAX,J)=0.
      HNRTH(IMAX,J)=0.
  110 Q(IMAX,J)=0.
C
C     THE UPPER BORDER, WHERE DPSI/DY=0.
C
      DO 120 I=IXWDTH,IMAX
      HNRTH(I,JMAX)=0.
  120 HSTH(I,JMAX)=2.E0*HSTH(I,JMAX)
C
C     THE LEFT BORDER, WHERE DPSI/DX=0.
C
```

```
      DO 130 J=1,JYDPTH
      HWEST(1,J)=0.
  130 HEAST(1,J)=2.E0*HEAST(1,J)
C
C     THE CORNER REGION
C
      X0=XPOS(IXWDTH)-RAD
      Y0=YPOS(JYDPTH)-RAD
      I=1
  140 J=JMAX+1
  150 J=J-1
      IF (J.LT.JYDPTH)GO TO 180
      IF (XPOS(I).GT.X0)GO TO 170
  160 HSTH(I,J)=0.
      HWEST(I,J)=0.
      HCNTR(I,J)=1.E0
      HEAST(I,J)=0.
      HNRTH(I,J)=0.
      Q(I,J)=1.0
      IDENT(I,J)=-1
      GO TO 150
  170 IF(YPOS(J).LE.Y0)GO TO 160
C
C     TEST TO DETERMINE IF THE POINT IS EXTERNAL TO THE CIRCLE.
C
      XP=XPOS(I)-X0
      YP=YPOS(J)-Y0
      TSTP=CIRCLE(XP,YP,RAD)
      IF(TSTP.LE.0.)GO TO 160
C
C     FIND THE WEST & NORTH DISTANCES TO THE BOUNDARY.  FIRST
C     DETERMINE WHICH POINTS ARE ALSO EXTERNAL, IF ANY.
C
      XW=XPOS(I-1)-X0
      YN=YPOS(J+1)-Y0
      TSTW=CIRCLE(XW,YP,RAD)
      TSTN=CIRCLE(XP,YN,RAD)
      XW=XPOS(I-1)
      IF(TSTW.LT.0.)XW=X0+POINT(YP,RAD)
      IF(TSTW.LT.0.) IDENT(I,J)=1
      XW=XPOS(I)-XW
      YN=YPOS(J+1)
      IF(TSTN.LT.0.)YN=Y0+POINT(XP,RAD)
      IF(TSTN.LT.0.) IDENT(I,J)=1
      YN=YPOS(J)-YN
      XWW=XDLT(I)/100.
      YNN=YDLT(J)/100.
      IF(XW.LT.XWW.OR.YN.LT.YNN) GO TO 160
C
C     MODIFY HWEST & HNRTH, TO ACCOUNT FOR THE CIRCULAR BOUNDARY.
C
      XAV=(XDLT(I+1)+XW)/2.E0
      YAV=(YN+YDLT(J))/2.E0
      HWEST(I,J)=YAV/XW
      HNRTH(I,J)=XAV/YN
      HCNTR(I,J)=-(HSTH(I,J)+HWEST(I,J)+HEAST(I,J)+HNRTH(I,J))
```

```
C
C       MODIFY QAREA TO ACCOUNT FOR THE PORTION (ASSUMED A
C       TRIANGLE) ECLIPSED BY THE CIRCLE.
C
        X1=XPOS(I)-XDLT(I)/2.E0
        Y1=YPOS(J)-YDLT(J+1)/2.E0
        XP=X1-X0
        YP=Y1-Y0
        TSTP=CIRCLE(XP,YP,RAD)
C
C       IF (TSTP.GE.0.) NO AREA IS ECLIPSED.
C
        IF(TSTP.GE.0.)GO TO 150
        Y2=POINT(XP,RAD)+Y0
        X3=POINT(YP,RAD)+X0
        QAREA(I,J)=QAREA(I,J)-(Y2-Y1)*(X3-X1)/2.E0
        GO TO 150
C
C       INDEX I & CONTINUE UNTIL I.GT.IXWDTH.
C
  180   I=I+1
        IF(I.LE.IXWDTH) GO TO 140
        IMAXX=IMAX-1
        DO 190 I=1,IMAXX
        DO 190 J=2,JMAX
  190   IF(IDENT(I,J).LT.0) PSI(I,J)=1.0
        RETURN
        END
C
C
```

```
      SUBROUTINE NITIAL
C
C     SUPPLIES INITIAL ESTIMATE FOR PSI(I,J), USING THE DEPLETION
C     APPROXIMATION
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,UBLTIN,UOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIOLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      RD=RAD+1.0
      I=1
C
C     THE HORIZONTAL SHEATH
C
  100 J=JYDPTH
  110 J=J-1
      IF(J.EQ.1) GO TO 120
      RR=YPOS(J)-Y0
      IF(RR.GE.RD) GO TO 120
      PSI(I,J)=(RD-RR)*(RD-RR)
      GO TO 110
  120 I=I+1
      IF(I.EQ.IMAX.OR.XPOS(I).GT.X0) GO TO 130
      GO TO 100
  130 IM=I-1
      J=JMAX
C
C     THE VERTICAL SHEATH
C
  140 I=IXWDTH
  150 I=I+1
      IF(I.EQ.IMAX) GO TO 160
      RR=XPOS(I)-X0
      IF(RR.GE.RD) GO TO 160
      PSI(I,J)=(RD-RR)*(RD-RR)
      GO TO 150
  160 J=J-1
      IF(J.EQ.1.OR.YPOS(J).GT.Y0) GO TO 170
      GO TO 140
C
C     THE CIRCULAR CORNER SHEATH
C
  170 I=IM
  180 I=I+1
      IF(I.EQ.IMAX) GO TO 190
      XP=XPOS(I)-X0
      YP=YPOS(J)-Y0
```

```
      RR=SQRT(XP*XP+YP*YP)
      IF(RR.LE.RAD) GO TO 180
      IF(RR.GE.RD) GO TO 190
      PSI(I,J)=(RD-RR)*(RD-RR)
      GO TO 180
  190 J=J-1
      IF(J.LE.1) RETURN
      YP=YPOS(J)-Y0
      IF(YP.GE.RD) RETURN
      GO TO 170
      END
C
C
```

```
      SUBROUTINE GRELAX
C
C     IMPLEMENTS GUMMEL'S RELAXATION ALGORITHM FOR PSI
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      OMEGA=1.6
      IT=1
      IMAXX=IMAX-1
  100 DLTMX=0
      CALL DNSITY
      CALL POISSN
      DO 110 I=1,IMAXX
      DO 110 J=2,JMAX
      DLTMX=AMAX1(ABS(DELTA(I,J)),DLTMX)
      PSI(I,J)=PSI(I,J)+OMEGA*DELTA(I,J)
      IF(PSI(I,J).LT.0.) PSI(I,J)=0.
  110 CONTINUE
      DMX=0.
      DO 500 I=1,IMAXX
      DO 500 J=2,JMAX
      IF(ABS(DELTA(I,J)).LE.DMX) GO TO 500
      DMX=ABS(DELTA(I,J))
      II=I
      JJ=J
  500 CONTINUE
      WRITE(6,5000) DELTA(II,JJ),II,JJ
 5000 FORMAT(1H0,5X,'MAX DELTA IN GRELAX IS',1PE11.3,1X,
     &'AT COORDINATE (',I2,',',I2,')')
      IF(DLTMX.LE.CONVRG) GO TO 120
      IT=IT+1
      IF(IT.GT.ITMAX) GO TO 130
      GO TO 100
  120 WRITE(6,1000) IT
 1000 FORMAT(1H0,5X,I5,'  ITERATIONS WERE REQUIRED IN GRELAX',//)
      RETURN
  130 WRITE(6,1010) DLTMX,CONVRG
 1010 FORMAT(1H0,5X,'MAXIMUM ITERATIONS WERE EXCEEDED IN
     &GRELAX.    DLTMX=',1PE10.2,'  CONVRG=',1PE10.2,//)
      STOP
      END
C
C
```

```
      SUBROUTINE POISSN
C
C     SOLVES POISSONS EQUATION USING STONES METHOD.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      DIMENZR:TZRGEW(31,31)
C
      DO 100 I=1,IMAX
      DO 100 J=1,JMAX
      PSINEW(I,J)=PSI(I,J)
      B(I,J)=HSTH(I,J)
      D(I,J)=HWEST(I,J)
      E(I,J)=HCNTR(I,J)
      F(I,J)=HEAST(I,J)
  100 H(I,J)=HNRTH(I,J)
      CALL CHARGE
C
C     ITERATIVE STONE'S METHOD
C
      ISTONE=0
      ITR=1
  110 ISTONE=ISTONE+1
      CALL STONE(ISTONE,IMAX,JMAX,RESID,PSINEW)
      IF(RESID.LE.SIDLT) GO TO 120
      ITR=ITR+1
      IF(ITR.LE.ITRMAX) GO TO 110
      WRITE(6,1000)
 1000 FORMAT(1H0,1X,'MAXIMUM ITERATIONS EXCEEDED IN POISSN',/)
      STOP
  120 DO 130 I=1,IMAX
      DO 130 J=1,JMAX
  130 DELTA(I,J)=PSINEW(I,J)-PSI(I,J)
      WRITE(6,5000) ITR
 5000 FORMAT(2X,//,2X,I5,1X,'ITERATIONS WERE REQUIRED WITH STONES METHOD
     &IN POISSN')
      RETURN
      END
C
C
```

```
      SUBROUTINE DNSITY
C
C     USES BOLTZMANN DISTRIBUTIONS FOR THE HOLE & ELECTRON DENSITIES.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      EFACTR=PSINRM/BLTZMN
      DO 130 I=1,IMAX
      DO 130 J=1,JMAX
      IF(IDENT(I,J).LT.0) GO TO 130
      EXPNT=PSI(I,J)*EFACTR
      IF(EXPNT.GT.80.) GO TO 100
      HDENS(I,J)=1.0/EXP(EXPNT)
      GO TO 110
  100 HDENS(I,J)=0.
  110 VOLT=PSINRM-VBLTIN
      EXPNT=2.*ALOG(TRNSIC/DNSNRM)+(PSINRM*PSI(I,J)-VOLT)/BLTZMN
      IF(EXPNT.LT.-80.) GO TO 120
      EDENS(I,J)=EXP(EXPNT)
      GO TO 130
  120 EDENS(I,J)=0.
  130 CONTINUE
      RETURN
      END
C
C
```

```
      SUBROUTINE CHARGE
C
C     CALCULATES THE CHARGE TERMS IN GUMMELS ALGORITHM
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      ALPHA=PSINRM/BLTZMN
      IMAXX=IMAX-1
      DO 100 I=1,IMAXX
      DO 100 J=2,JMAX
      IF(IDENT(I,J).LT.0) GO TO 100
      Q(I,J)=1.+EDENS(I,J)-HDENS(I,J)-ALPHA*(EDENS(I,J)+HDENS(I,J))*
     &PSI(I,J)
      Q(I,J)=2.*Q(I,J)*QAREA(I,J)
      E(I,J)=E(I,J)-2.*(EDENS(I,J)+HDENS(I,J))*ALPHA*QAREA(I,J)
  100 CONTINUE
C
C     MODIFY Q AT THE UPPER BORDER TO INCLUDE SURFACE CHARGE QSS.
C
      II=IXWDTH+1
      DO 110 I=II,IMAX
  110 Q(I,JMAX)=Q(I,JMAX)-QSS*(XDLT(I)+XDLT(I+1))
      RETURN
      END
C
C
```

```
      SUBROUTINE STONE(ISTONE,JMAX,KMAX,RESID,T)
C
C     IMPLEMENTS STONE'S METHOD FOR MATRIX FACTORIZATION.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      DIMENSION T(JMAX,KMAX)
      DIMENSION EE(31,31),FF(31,31),V(31,31),ALPHA(18)
C
C     DEFINE FREQUENTLY USED FUNCTIONS.(N FOR FORWARD K,R FOR REVERSE).
C
      BN(BDUMMY,EDUMMY)=BDUMMY/(1.E0+ALPH*EDUMMY)
      CN(DDUMMY,FDUMMY)=DDUMMY/(1.E0+ALPH*FDUMMY)
      DN1(EDUMMY,FDUMMY)=BB*(ALPH*EDUMMY-FDUMMY)
      DN2(FDUMMY,EDUMMY)=CC*(ALPH*FDUMMY-EDUMMY)
      EN(FDUMMY,EDUMMY)=(FDUMMY-ALPH*BB*EDUMMY)/DD
      FN(HDUMMY,FDUMMY)=(HDUMMY-ALPH*CC*FDUMMY)/DD
      BR(HDUMMY,EDUMMY)=HDUMMY/(1.E0+ALPH*EDUMMY)
      CR(DDUMMY,FDUMMY)=DDUMMY/(1.E0+ALPH*FDUMMY)
      DR1(EDUMMY,FDUMMY)=BB*(ALPH*EDUMMY-FDUMMY)
      DR2(FDUMMY,EDUMMY)=CC*(ALPH*FDUMMY-EDUMMY)
      ER(FDUMMY,EDUMMY)=(FDUMMY-ALPH*BB*EDUMMY)/DD
      FR(BDUMMY,FDUMMY)=(BDUMMY-ALPH*CC*FDUMMY)/DD
C
C     ISTONE FLAGS WHICH ALPHA TO USE AND WHETHER FORWARD (ISTONE ODD)
C     OR REVERSE (ISTONE EVEN) INDEXING OF K IS TO BE IMPLEMENTED.
C
      IF(ISTONE.GT.18) ISTONE=1
      ALPH=ALPHA(ISTONE)
      JMAXX=JMAX-1
      KMAXX=KMAX-1
      J=2*(ISTONE/2)
      IF(J.EQ.ISTONE) GO TO 170
C
C     ISTONE IS ODD.  INDEX K FORWARD
C
C
C     THE CORNER POINT (1,1)
C
      DD=E(1,1)
      EE(1,1)=F(1,1)/DD
      FF(1,1)=H(1,1)/DD
      R=Q(1,1)-(E(1,1)*T(1,1)+F(1,1)*T(2,1)+H(1,1)*T(1,2))
      RR=ABS(R)
      RESID=RR
      V(1,1)=R/DD
```

```
      IF(ABS(V(1,1)).LT.1.E-7) V(1,1)=0.
C
C     THE ROW K=1, EXCLUDING THE RIGHT CORNER POINT.
C
      DO 100 J=2,JMAXX
      JP=J+1
      JN=J-1
      CC=CN(D(J,1),FF(JN,1))
      DD=E(J,1)+DN2(FF(JN,1),EE(JN,1))
      EE(J,1)=F(J,1)/DD
      FF(J,1)=FN(H(J,1),FF(JN,1))
      R=Q(J,1)-(D(J,1)*T(JN,1)+E(J,1)*T(J,1)+F(J,1)*T(JP,1)+H(J,1)*
     &T(J,2))
      RR=ABS(R)
      IF (RR.GT.RESID) RESID=RR
      V(J,1)=(R-CC*V(JN,1))/DD
  100 IF(ABS(V(J,1)).LT.1.E-7) V(J,1)=0.
C
C     THE CORNER POINT (JMAX,1)
C
      J=JMAX
      JN=JMAXX
      CC=CN(D(J,1),FF(JN,1))
      DD=E(J,1)+DN2(FF(JN,1),EE(JN,1))
      EE(J,1)=0.
      FF(J,1)=FN(H(J,1),FF(JN,1))
      R=Q(J,1)-(D(J,1)*T(JN,1)+E(J,1)*T(J,1)+H(J,1)*T(J,2))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(J,1)=(R-CC*V(JN,1))/DD
      IF(ABS(V(J,1)).LT.1.E-7) V(J,1)=0.
C
C     ALL ROWS EXCEPT ROW KMAX
C
      DO 120 K=2,KMAXX
      KP=K+1
      KN=K-1
C
C     THE LEFT BORDER POINT
C
      BB=BN(B(1,K),EE(1,KN))
      DD=E(1,K)+DN1(EE(1,KN),FF(1,KN))
      EE(1,K)=EN(F(1,K),EE(1,KN))
      FF(1,K)=H(1,K)/DD
      R=Q(1,K)-(B(1,K)*T(1,KN)+E(1,K)*T(1,K)+F(1,K)*T(2,K)+H(1,K)*
     &T(1,KP))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(1,K)=(R-BB*V(1,KN))/DD
      IF(ABS(V(1,K)).LT.1.E-7) V(1,K)=0.
C
C     THE ROW K EXCLUDING THE RIGHT BORDER POINT.
C
      DO 110 J=2,JMAXX
      JP=J+1
      JN=J-1
```

```
      BB=BN(B(J,K),EE(J,KN))
      CC=CN(D(J,K),FF(JN,K))
      DD=E(J,K)+DN1(EE(J,KN),FF(J,KN))+DN2(FF(JN,K),EE(JN,K))
      EE(J,K)=EN(F(J,K),EE(J,KN))
      FF(J,K)=FN(H(J,K),FF(JN,K))
      R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+
     &F(J,K)*T(JP,K)+H(J,K)*T(J,KP))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      U(J,K)=(R-BB*U(J,KN)-CC*U(JN,K))/DD
  110 IF(ABS(U(J,K)).LT.1.E-7) U(J,K)=0.
C
C     THE RIGHT BORDER POINT
C
      J=JMAX
      JN=JMAXX
      BB=BN(B(J,K),EE(J,KN))
      CC=CN(D(J,K),FF(JN,K))
      DD=E(J,K)+DN1(EE(J,KN),FF(J,KN))+DN2(FF(JN,K),EE(JN,K))
      EE(J,K)=0
      FF(J,K)=FN(H(J,K),FF(JN,K))
      R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+H(J,K)*
     &T(J,KP))
      RR=ABS(R)
      IF (RR.GT.RESID) RESID=RR
      U(J,K)=(R-BB*U(J,KN)-CC*U(JN,K))/DD
  120 IF(ABS(U(J,K)).LT.1.E-7) U(J,K)=0.
C
C     THE CORNER POINT (1,KMAX)
C
      K=KMAX
      KN=KMAXX
      BB=BN(B(1,K),EE(1,KN))
      DD=E(1,K)+DN1(EE(1,KN),FF(1,KN))
      EE(1,K)=EN(F(1,K),EE(1,KN))
      FF(1,K)=0
      R=Q(1,K)-(B(1,K)*T(1,KN)+E(1,K)*T(1,K)+F(1,K)*T(2,K))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      U(1,K)=(R-BB*U(1,KN))/DD
      IF(ABS(U(1,K)).LT.1.E-7) U(1,K)=0.
C
C     THE ROW KMAX, EXCLUDING THE CORNER POINT (JMAX,KMAX)
C
      DO 130 J=2,JMAXX
      JP=J+1
      JN=J-1
      BB=BN(B(J,K),EE(J,KN))
      CC=CN(D(J,K),FF(JN,K))
      DD=E(J,K)+DN1(EE(J,KN),FF(J,KN))+DN2(FF(JN,K),EE(JN,K))
      EE(J,K)=EN(F(J,K),EE(J,KN))
      FF(J,K)=0
      R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+F(J,K)*
     &T(JP,K))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
```

```
      U(J,K)=(R-BB*U(J,KN)-CC*U(JN,K))/DD
  130 IF(ABS(U(J,K)).LT.1.E-7) U(J,K)=0.
C
C     THE CORNER POINT (JMAX,KMAX)
C
      J=JMAX
      JN=JMAXX
      BB=BN(B(J,K),EE(J,KN))
      CC=CN(D(J,K),FF(JN,K))
      DD=E(J,K)+DN1(EE(J,KN),FF(J,KN))+DN2(FF(JN,K),EE(JN,K))
      EE(J,K)=0
      FF(J,K)=0
      R=R(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      U(J,K)=(R-BB*U(J,KN)-CC*U(JN,K))/DD
      IF(ABS(U(J,K)).LT.1.E-7) U(J,K)=0.
C
C     INDEX K AND J IN REVERSE TO FIND DELTA
C
C     THE CORNER POINT (JMAX,KMAX)
C
      K=KMAX
      DELTA(JMAX,K)=U(JMAX,K)
C
C     THE ROW KMAX
C
      DO 140 JJ=1,JMAXX
      J=JMAXX+1-JJ
      JP=J+1
      DELTA(J,K)=U(J,K)-EE(J,K)*DELTA(JP,K)
  140 IF(ABS(DELTA(J,K)).LT.1.E-10) DELTA(J,K)=0.
C
C     THE ROW K. TREATING THE RIGHT BORDER FIRST.
C
      DO 150 KK=1,KMAXX
      K=KMAXX+1-KK
      KP=K+1
      DELTA(JMAX,K)=U(JMAX,K)-FF(JMAX,K)*DELTA(JMAX,KP)
      IF(ABS(DELTA(JMAX,K)).LT.1.E-10) DELTA(JMAX,K)=0.
      DO 150 JJ=1,JMAXX
      J=JMAXX+1-JJ
      JP=J+1
      DELTA(J,K)=U(J,K)-EE(J,K)*DELTA(JP,K)-FF(J,K)*DELTA(J,KP)
  150 IF(ABS(DELTA(J,K)).LT.1.E-10) DELTA(J,K)=0.
      GO TO 240
C
C     ISTONE IS EVEN.  INDEX K IN REVERSE ORDER.
C
C     THE CORNER POINT (1,KMAX)
C
  170 K=KMAX
      KN=KMAXX
      DD=E(1,K)
      EE(1,K)=F(1,K)/DD
      FF(1,K)=B(1,K)/DD
```

```
      R=Q(1,K)-(B(1,K)*T(1,KN)+E(1,K)*T(1,K)+F(1,K)*T(2,K))
      RR=ABS(R)
      RESID=RR
      U(1,K)=R/DD
      IF(ABS(U(1,K)).LT.1.E-7) U(1,K)=0.
C
C     THE ROW KMAX
C
      DO 180 J=2,JMAXX
      JP=J+1
      JN=J-1
      CC=CR(D(J,K),FF(JN,K))
      DD=E(J,K)+DR2(FF(JN,K),EE(JN,K))
      EE(J,K)=F(J,K)/DD
      FF(J,K)=FR(B(J,K),FF(JN,K))
      R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+F(J,K)*
     &T(JP,K))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      U(J,K)=(R-CC*U(JN,K))/DD
  180 IF(ABS(U(J,K)).LT.1.E-7) U(J,K)=0.
C
C     THE CORNER POINT (JMAX,KMAX)
C
      J=JMAX
      JN=JMAXX
      CC=CR(D(J,K),FF(JN,K))
      DD=E(J,K)+DR2(FF(JN,K),EE(JN,K))
      EE(J,K)=0
      FF(J,K)=FR(B(J,K),FF(JN,K))
      R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      U(J,K)=(R-CC*U(JN,K))/DD
      IF(ABS(U(J,K)).LT.1.E-7) U(J,K)=0.
C
C     ALL ROWS EXCEPT THE ROW K=1
C
      DO 200 KK=2,KMAXX
      K=KMAXX+2-KK
      KP=K+1
      KN=K-1
C
C     THE LEFT BORDER POINT (1,K)
C
      BB=BR(H(1,K),EE(1,KP))
      DD=E(1,K)+DR1(EE(1,KP),FF(1,KP))
      EE(1,K)=ER(F(1,K),EE(1,KP))
      FF(1,K)=B(1,K)/DD
      R=Q(1,K)-(B(1,K)*T(1,KN)+E(1,K)*T(1,K)+F(1,K)*T(2,K)+H(1,K)*
     &T(1,KP))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      U(1,K)=(R-BB*U(1,KP))/DD
      IF(ABS(U(1,K)).LT.1.E-7) U(1,K)=0.
C
```

```
C       THE ROW K EXCLUDING THE RIGHT BORDER POINT
C
        DO 190 J=2,JMAXX
        JP=J+1
        JN=J-1.
        BB=BR(H(J,K),EE(J,KP))
        CC=CR(D(J,K),FF(JN,K))
        DD=E(J,K)+DR1(EE(J,KP),FF(J,KP))+DR2(FF(JN,K),EE(JN,K))
        EE(J,K)=ER(F(J,K),EE(J,KP))
        FF(J,K)=FR(B(J,K),FF(JN,K))
        R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+F(J,K)*
       &T(JP,K)+H(J,K)*T(J,KP))
        RR=ABS(R)
        IF(RR.GT.RESID) RESID=RR
        U(J,K)=(R-BB*U(J,KP)-CC*U(JN,K))/DD
  190   IF(ABS(U(J,K)).LT.1.E-7) U(J,K)=0.
C
C       THE RIGHT BORDER POINT
C
        J=JMAX
        JN=JMAXX
        BB=BR(H(J,K),EE(J,KP))
        CC=CR(D(J,K),FF(JN,K))
        DD=E(J,K)+DR1(EE(J,KP),FF(J,KP))+DR2(FF(JN,K),EE(JN,K))
        EE(J,K)=0.
        FF(J,K)=FR(B(J,K),FF(JN,K))
        R=Q(J,K)-(B(J,K)*T(J,KN)+D(J,K)*T(JN,K)+E(J,K)*T(J,K)+H(J,K)*
       &T(J,KP))
        RR=ABS(R)
        IF(RR.GT.RESID) RESID=RR
        U(J,K)=(R-BB*U(J,KP)-CC*U(JN,K))/DD
  200   IF(ABS(U(J,K)).LT.1.E-7) U(J,K)=0.
C
C       THE CORNER POINT (1,1)
C
        BB=BR(H(1,1),EE(1,2))
        DD=E(1,1)+DR1(EE(1,2),FF(1,2))
        EE(1,1)=ER(F(1,1),EE(1,2))
        FF(1,1)=0.
        R=Q(1,1)-(E(1,1)*T(1,1)+F(1,1)*T(2,1)+H(1,1)*T(1,2))
        RR=ABS(R)
        IF(RR.GT.RESID) RESID=RR
        U(1,1)=(R-BB*U(1,2))/DD
        IF(ABS(U(1,1)).LT.1.E-7) U(1,1)=0.
C
C       THE ROW K=1, EXCLUDING THE RIGHT CORNER POINT.
C
        DO 210 J=2,JMAXX
        JP=J+1
        JN=J-1
        BB=BR(H(J,1),EE(J,2))
        CC=CR(D(J,1),FF(JN,1))
        DD=E(J,1)+DR1(EE(J,2),FF(J,2))+DR2(FF(JN,1),EE(JN,1))
        EE(J,1)=ER(F(J,1),EE(J,2))
        FF(J,1)=0.
        R=Q(J,1)-(D(J,1)*T(JN,1)+E(J,1)*T(J,1)+F(J,1)*T(JP,1)+H(J,1)*
       &T(J,2))
```

```
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(J,1)=(R-BB*V(J,2)-CC*V(JN,1))/DD
  210 IF(ABS(V(J,1)).LT.1.E-7) V(J,1)=0.
C
C     THE CORNER POINT (JMAX,1)
C
      J=JMAX
      JN=JMAXX
      BB=BR(H(J,1),EE(J,2))
      CC=CR(D(J,1),FF(JN,1))
      DD=E(J,1)+DR1(EE(J,2),FF(J,2))+DR2(FF(JN,1),EE(JN,1))
      EE(J,1)=0.
      FF(J,1)=0.
      R=B(J,1)-(D(J,1)*T(JN,1)+E(J,1)*T(J,1)+H(J,1)*T(J,2))
      RR=ABS(R)
      IF(RR.GT.RESID) RESID=RR
      V(J,1)=(R-BB*V(J,2)-CC*V(JN,1))/DD
      IF(ABS(V(J,1)).LT.1.E-7) V(J,1)=0.
C
C     INDEX K FORWARD & J REVERSE TO FIND DELTA
C
C     THE POINT (JMAX,1)
C
      DELTA(JMAX,1)=V(JMAX,1)
C
C     THE ROW K=1
C
      DO 220 JJ=1,JMAXX
      J=JMAXX+1-JJ
      JP=J+1
      DELTA(J,1)=V(J,1)-EE(J,1)*DELTA(JP,1)
  220 IF(ABS(DELTA(J,1)).LT.1.E-10) DELTA(J,1)=0.
C
C     ALL REMAINING ROWS
C
      DO 230 K=2,KMAX
      KN=K-1
C
C     THE RIGHT BORDER POINT
C
      DELTA(JMAX,K)=V(JMAX,K)-FF(JMAX,K)*DELTA(JMAX,KN)
      IF(ABS(DELTA(JMAX,K)).LT.1.E-10) DELTA(JMAX,K)=0.
C
C     ALL REMAINING POINTS OF THE ROW
C
      DO 230 JJ=1,JMAXX
      J=JMAXX+1-JJ
      JP=J+1
      DELTA(J,K)=V(J,K)-EE(J,K)*DELTA(JP,K)-FF(J,K)*DELTA(J,KN)
  230 IF(ABS(DELTA(J,K)).LT.1.E-10) DELTA(J,K)=0.
C
C     CALCULATE NEW T(J,K)
C
  240 DO 250 J=1,JMAX
      DO 250 K=1,KMAX
```

```
  250 T(J,K)=T(J,K)+DELTA(J,K)
      RETURN
C
C     ENTRY STONE1 SETS UP ALPHA VALUES & INITIALIZES ALL ARRAYS
C
      ENTRY STONE1(JMAX,KMAX)
C
      DO 320 J=2,JMAX
      DO 320 K=2,KMAX
      DX=XDLT(J)/XPOS(JMAX)
      DY=YDLT(K)/YPOS(1)
  320 EE(J,K)=2.E0/(1.E0/(DX*DX)+1.E0/(DY*DY))
C
C     FIND AVERAGE
C
      ALPHX=0
      DO 330 J=2,JMAX
      DO 330 K=2,KMAX
  330 ALPHX=ALPHX+EE(J,K)
      ALPHX=ALPHX/(JMAX*KMAX)
C
C     IMPLEMENT STONE'S EQ(27)
C
      DO 340 J=1,9
      ALPHY=J-1
      ALPHY=ALPHY/8.E0
  340 EE(J,1)=ALPHX**ALPHY
C
C     IMPLEMENT THE SEQUENCE OF 18 ALPHAS SUGGESTED BY STONE
C
      ALPHA(1)=EE(9,1)
      ALPHA(3)=EE(6,1)
      ALPHA(5)=EE(3,1)
      ALPHA(7)=EE(8,1)
      ALPHA(9)=EE(5,1)
      ALPHA(11)=EE(2,1)
      ALPHA(13)=EE(7,1)
      ALPHA(15)=EE(4,1)
      ALPHA(17)=EE(1,1)
      DO 350 J=2,18,2
  350 ALPHA(J)=ALPHA(J-1)
      DO 360 J=1,18
  360 ALPHA(J)=1.E0-ALPHA(J)
C
C
C     INITIALIZE ALL ARRAYS INTERNAL TO STONE TO ZERO.
C
      DO 370 J=1,JMAX
      DO 370 K=1,KMAX
      EE(J,K)=0.
      FF(J,K)=0.
      V(J,K)=0.
  370 DELTA(J,K)=0.
      RETURN
      END
C
```

```
      SUBROUTINE EFIELD
C
C     CALCULATES THE ELECTRIC FIELD COMPONENTS ESUBX & ESUBY
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
C     ESUBX COMPONENT
C
      IMAXX=IMAX-1
      DO 140 I=2,IMAXX
      DO 140 J=2,JMAX
      XW=XDLT(I)
      IF(IDENT(I,J).LT.0) GO TO 140
      IF(IDENT(I,J).EQ.0) GO TO 120
      IF(J.EQ.JMAX) GO TO 100
      YAV=(YDLT(J+1)+YDLT(J))/2.E0
      GO TO 110
  100 YAV=YDLT(J)
  110 XW=YAV/HWEST(I,J)
  120 ESUBX(I,J)=((PSI(I,J)-PSI(I+1,J))/XDLT(I+1)+(PSI(I-1,J)-PSI(I,J))
     &/XW)/2.E0
  140 CONTINUE
C
C     ESUBY COMPONENT
C
      JMAXX=JMAX-1
      DO 190 I=1,IMAXX
      DO 190 J=2,JMAXX
      YN=YDLT(J+1)
      IF(IDENT(I,J).LT.0) GO TO 190
      IF(IDENT(I,J).EQ.0) GO TO 170
      IF(I.EQ.1) GO TO 150
      XAV=(XDLT(I)+XDLT(I+1))/2.E0
      GO TO 160
  150 XAV=XDLT(2)
  160 YN=XAV/HNRTH(I,J)
  170 ESUBY(I,J)=((PSI(I,J+1)-PSI(I,J))/YN+(PSI(I,J)-PSI(I,J-1))/
     &YDLT(J))/2.E0
  190 CONTINUE
C
C     APPLY QUADRATIC EXTRAPOLATION TO FILL IN ESUBX & ESUBY AT THE
C     EXTERNAL POINTS BORDERING THE INTERFACE.
C
      DO 200 I=1,IMAX
      DO 200 J=1,JMAX
```

```
      IF(IDENT(I,J).GE.0) GO TO 200
      IF(IDENT(I+1,J).LT.0) GO TO 200
      DENOM=XDLT(I+3)*XDLT(I+2)*(XDLT(I+3)+XDLT(I+2))
      XX=XDLT(I+2)+XDLT(I+1)
      DENUM=XDLT(I+2)*ESUBX(I+3,J)+XDLT(I+3)*ESUBX(I+1,J)-
     &(XDLT(I+2)+XDLT(I+3))*ESUBX(I+2,J)
      AA=DENUM/DENOM
      DENUM=XDLT(I+2)*XDLT(I+2)*ESUBX(I+3,J)-XDLT(I+3)*XDLT(I+3)*
     &ESUBX(I+1,J)-(XDLT(I+2)*XDLT(I+2)-XDLT(I+3)*XDLT(I+3))*
     &ESUBX(I+2,J)
      BB=DENUM/DENOM
      ESUBX(I,J)=ESUBX(I+2,J)+AA*XX*XX-BB*XX
      DENOM=YDLT(J-2)*YDLT(J-1)*(YDLT(J-2)+YDLT(J-1))
      XX=YDLT(J-1)+YDLT(J)
      DENUM=YDLT(J-1)*ESUBY(I,J-3)+YDLT(J-2)*ESUBY(I,J-1)-
     &(YDLT(J-1)+YDLT(J-2))*ESUBY(I,J-2)
      AA=DENUM/DENOM
      DENUM=YDLT(J-1)*YDLT(J-1)*ESUBY(I,J-3)-YDLT(J-2)*YDLT(J-2)*
     &ESUBY(I,J-1)-(YDLT(J-1)*YDLT(J-1)-YDLT(J-2)*YDLT(J-2))*
     &ESUBY(I,J-2)
      BB=DENUM/DENOM
      ESUBY(I,J)=ESUBY(I,J-2)+AA*XX*XX-BB*XX
  200 CONTINUE
      RETURN
      END
C
C
```

```
      SUBROUTINE AVLNCH(VLTOLT,AVOLT,XPATH,YPATH,ET,KNTMAX)
C
C
C     EVALUATES THE BREAKDOWN VOLTAGE AFTER LOCATING THE TRAJECTORY
C     ALONG WHICH THE IONIZATION INTEGRAL IS A MAXIMUM.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XOLT(32),YOLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIOLT
      COMMON/NRMLZE/ONSNRM,PSINRM,DSTNRM,BLTZMN
      COMMON/RSCALE/ RSFCTR
C
      DIMENSION XSTART(3),YSTART(3),PI(3),KNTMX(3),ISTART(3),JSTART(3)
      DIMENSION XPATH(KNTMAX,3),YPATH(KNTMAX,3),ET(KNTMAX,3)
C
C     CALCULATE EQUILIBRIUM PARAMETERS. ESTIMATE THE BREAKDOWN VOLTAGE,
C     AND CALCULATE INITIAL NORMALIZATIONS.
C
      CALL EQULIB
      CALL VGUESS(VOLT)
      WRITE(6,900) VOLT
  900 FORMAT(2X,//,2X,'THE INITIAL VOLTAGE ESTIMATE IS',1PE10.2,
     &1X,'VOLTS',//)
      RSFCTR=1.0
      RDSAVE=RAD
      XWSAVE=XWIDTH
      YDSAVE=YDEPTH
      QSSAVE=QSS
      PSINRM=VBLTIN+VOLT
      DSTNRM=SQRT(2.*11.7*8.854E-14*PSINRM/(1.602E-19*ONSNRM))
      KSWCH=0
      K=1
      ISTART(2)=IXWDTH
      JSTART(2)=JYDPTH
C
C     PROVIDE CURRENT NORMALIZATIONS OF RAD,XWIDTH,YDEPTH, & QSS
C
  100 RAD=RDSAVE/(DSTNRM*1.E4)
      XWIDTH=XWSAVE/(DSTNRM*1.E4)
      YDEPTH=YDSAVE/(DSTNRM*1.E4)
      QSS=QSSAVE*DSTNRM/(11.7*8.854E-14*PSINRM)
      SINRM0=PSINRM
C
C     USING PRESENT NORMALIZATIONS, SOLVE POISSON'S EQ. & OBTAIN THE
C     ELECTRIC FIELD LATTICES. IF KSWCH.EQ.0, LOCATE POSITION OF MAX
C     E-FIELD AND ESTABLISH THREE STARTING LOCATIONS CENTERED ABOUT THAT
C     POSITION. IF KSWCH.GT.0, ESTABLISH ONLY THE CENTRAL STARTING
C     LOCATION.
```

```
C
      CALL GRID
      CALL STONE1(IMAX,JMAX)
      CALL BORDER
      CALL NITIAL
      CALL GRELAX
      CALL EFIELD
      I=ISTART(2)-1
      J=JSTART(2)-1
      IF(RAD.NE.XWIDTH.OR.RAD.NE.YDEPTH.OR.QSS.NE.0.) GO TO 105
      K=2
      KSWCH=1
      GO TO 120
  105 IF(KSWCH.GT.0) GO TO 120
      ETST=2.E5*DSTNRM/PSINRM
      EMAX=0.
      DO 110 II=1,IMAX
      DO 110 JJ=1,JMAX
      ETT=SQRT(ESUBX(II,JJ)*ESUBX(II,JJ)+ESUBY(II,JJ)*ESUBY(II,JJ))
      IF(ETT.LE.EMAX) GO TO 110
      EMAX=ETT
      I=II
      J=JJ
  110 CONTINUE
      IF(EMAX.GE.ETST) GO TO 115
      PSINRM=PSINRM*1.05*(ETST/EMAX)*(ETST/EMAX)
      DSTNRM=SQRT(2.*11.7*8.854E-14*PSINRM/(1.602E-19*DNSNRM))
  115 I=I-2
      J=J-2
C
C     ESTABLISH STARTING COORDINATES FOR TRAJECTORY K.
C
  120 I=I+1
      J=J+1
      X=XPOS(I)
      Y=YPOS(J)
      IF(X.GT.X0) GO TO 125
      XSTART(K)=X
      YSTART(K)=YPOS(JYDPTH)
      GO TO 150
  125 IF(Y.GT.Y0) GO TO 130
      XSTART(K)=XPOS(IXWDTH)
      YSTART(K)=Y
      GO TO 150
  130 X=X-X0
      Y=Y-Y0
      IF(X.LT.Y) GO TO 140
      Y=Y/X
      XSTART(K)=X0+RAD/SQRT(1.E0+Y*Y)
      YSTART(K)=Y0+(XSTART(K)-X0)*Y
      GO TO 150
  140 X=X/Y
      YSTART(K)=Y0+RAD/SQRT(1.E0+X*X)
      XSTART(K)=X0+(YSTART(K)-Y0)*X
  150 IF(KSWCH.GT.0) GO TO 180
      K=K+1
```

```
      IF(K.GT.3) GO TO 170
      GO TO 120
C
C     TRACE OUT THE TRAJECTORY FOR STARTING POSITION K.
C
  170 K=1
  180 KNT=1
      KEXIT=0
      X=XSTART(K)
      Y=YSTART(K)
      I=IXWDTH
      J=JYDPTH
      DLTX=X-XPOS(IXWDTH)
      DLTY=Y-YPOS(JYDPTH)
      CALL  INDEX(I,J,DLTX,DLTY,KEXIT)
      ISTART(K)=I
      JSTART(K)=J
      IF(DLTX.LT.XDLT(I+1)/1.E3) DLTX=XDLT(I+1)/1.E3
      IF(DLTY.LT.YDLT(J)/1.E3) DLTY=YDLT(J)/1.E3
      X=XPOS(I)+DLTX
      Y=YPOS(J)+DLTY
      CALL EFORCE(I,J,X,Y,DLTX,DLTY,EX,EY,ET(1,K))
      XPATH(1,K)=X
      YPATH(1,K)=Y
      ETT=ET(1,K)
  190 CALL TRAJEC(KNT,I,J,X,Y,DLTX,DLTY,ETT,KEXIT)
      IF(KEXIT.GT.0) GO TO 200
      KNT=KNT+1
      IF(KNT.GT.KNTMAX) GO TO 195
      XPATH(KNT,K)=X
      YPATH(KNT,K)=Y
      ET(KNT,K)=ETT
      GO TO 190
  195 WRITE(6,1000) KNTMAX
 1000 FORMAT(1H1,30X,/,5X,'EXCEEDED STORAGE ALLOTTED FOR TRAJECTORIES
     &'(KNTMAX=',I3,').',/,5X,'INCREASE THE VALUE OF THAT PARAMETER ',
     &'IN NAMELIST CONTRL.',/,5X,'INCREASE DIMENSIONS ',
     &'ALLOTED TO XPATH,YPATH,& ET IN MAIN,IF NECESSARY.')
      STOP
  200 KNTMX(K)=KNT
      KNT=1
      PI(K)=0.
      XST=XSTART(K)*DSTNRM*1.E4
      YST=YSTART(K)*DSTNRM*1.E4
      XPT=XPATH(1,K)*DSTNRM*1.E4
      YPT=YPATH(1,K)*DSTNRM*1.E4
      EPT=ET(1,K)*PSINRM/DSTNRM
      WRITE(6,1010)K,ISTART(K),JSTART(K),XST,YST,XPT,YPT,EPT,PI(K)
 1010 FORMAT(2X,//,2X,'K',3X,'ISTART',1X,'JSTART',3X,'XSTART',4X,
     &'YSTART',4X,'XPATH',5X,'YPATH',6X,'ET',7X,'ALPHA',5X,'PI',//,
     &1X,I2,5X,I2,5X,I2,3X,1P5E10.2,10X,1PE10.2)
C
C     EVALUATE THE IONIZATION INTO(ZPF ALONG TRAJECTORY K.
C
  210 KNT=KNT+1
      IF(KNT.GT.KNTMX(K)) GO TO 220
```

```
      DX=XPATH(KNT,K)-XPATH(KNT-1,K)
      DY=YPATH(KNT,K)-YPATH(KNT-1,K)
      ET2=ET(KNT,K)
      ET1=ET(KNT-1,K)
      PI0=PI(K)
      CALL INTGRL(DX,DY,ET1,ET2,PI(K),ALPHA)
      XPT=XPATH(KNT,K)*OSTNRM*1.E4
      YPT=YPATH(KNT,K)*OSTNRM*1.E4
      EPT=ET(KNT,K)*PSINRM/OSTNRM
      WRITE(6,1020)XPT,YPT,EPT,ALPHA,PI(K)
 1020 FORMAT(40X,1P5E10.2)
      TST=(PI(K)-PI0)/PI(K)
      IF(TST.GT.1.E-4) GO TO 210
  220 IF(KSWCH.GT.0) GO TO 230
      K=K+1
      IF(K.LT.4) GO TO 180
      KSWCH=1
  230 IF(K.EQ.2) GO TO 290
C
C     ADJUST STARTING POSITIONS UNTIL THE TRAJECTORY K=2 PRODUCES THE
C     MAXIMUM IONIZATION INTEGRAL.
C
      IF(PI(1).GT.PI(2)) GO TO 250
      IF(K.EQ.1) GO TO 240
      IF(PI(3).GT.PI(2)) GO TO 270
  240 K=2
      GO TO 290
C
C     MAXIMUM IS TO LEFT OF K=2. SHIFT LEFT ACCORDINGLY.
C
  250 K=1
      ISTART(2)=ISTART(1)
      JSTART(2)=JSTART(1)
      XSTART(2)=XSTART(1)
      YSTART(2)=YSTART(1)
      PI(2)=PI(1)
      KNTMX(2)=KNTMX(1)
      KMX=KNTMX(1)
      DO 260 KNT=1,KMX
      XPATH(KNT,2)=XPATH(KNT,1)
      YPATH(KNT,2)=YPATH(KNT,1)
  260 ET(KNT,2)=ET(KNT,1)
      I=ISTART(1)-2
      J=JSTART(1)-2
      IF(I.LT.0) GO TO 240
      GO TO 120
C
C     MAXIMUM IS TO RIGHT OF K=2. SHIFT RIGHT ACCORDINGLY.
C
  270 K=3
      ISTART(2)=ISTART(3)
      JSTART(2)=JSTART(3)
      XSTART(2)=XSTART(3)
      YSTART(2)=YSTART(3)
      PI(2)=PI(3)
      KNTMX(2)=KNTMX(3)
```

```
      KMX=KNTMX(3)
      DO 280 KNT=1,KMX
      XPATH(KNT,2)=XPATH(KNT,3)
      YPATH(KNT,2)=YPATH(KNT,3)
  280 ET(KNT,2)=ET(KNT,3)
      I=ISTART(3)
      J=JSTART(3)
      IF(J.EQ.JMAX) GO TO 240
      GO TO 120
C
C     ADJUST PSINRM UNTIL ONE MINUS PI CHANGES SIGN.  THEN USE THE
C     METHOD OF SECANTS TO FIND THE VALUE OF PSINRM LEADING TO
C     BREAKDOWN.
C
  290 Y1=1.0-PI(2)
      X1=PSINRM
      KEY=1
      IF(Y1.GT.0.) GO TO 300
      FCTR=0.9E0
      GO TO 310
  300 FCTR=1.1E0
  310 X2=X1*FCTR
      GO TO 360
  320 Y2=1.0-PPI
      TST=Y1*Y2
      IF(TST.LT.0.) GO TO 330
      Y1=Y2
      X1=X2
      GO TO 310
C
C     METHOD OF SECANTS.
C
  330 KEY=2
      KSEC=0
  340 DEN=Y2-Y1
      IF(ABS(Y2).LT.AVDLT) GO TO 380
      KSEC=KSEC+1
      XX=(X1*Y2-X2*Y1)/DEN
      X1=X2
      Y1=Y2
      X2=XX
      GO TO 360
  350 Y2=1.0-PPI
      GO TO 340
C
C     SUBPROGRAM FOR THE IONIZATION INTEGRAL
C
  360 KNT=1
      PPI=0.
      PSINRM=X2
      DSTNRM=DSTNRM*SQRT(X2/X1)
      WRITE(6,1030) KEY,PSINRM,DSTNRM
 1030 FORMAT(1H0,2X,'KEY=',I2,6X,'PSINRM=',1PE11.3,3X,'DSTNRM=',
     &1PE11.3,//)
      WRITE(6,1040)
 1040 FORMAT(1H0,2X,'KNT',4X,'DX',8X,'DY',8X,'ET1',7X,'ET2',
     &7X,'ALPHA',5X,'PI',//)
```

```
  370 KNT=KNT+1
      PPI0=PPI
      IF(KNT.GT.KNTMX(2)) GO TO (320,350),KEY
      DX=XPATH(KNT,2)-XPATH(KNT-1,2)
      DY=YPATH(KNT,2)-YPATH(KNT-1,2)
      ET2=ET(KNT,2)
      ET1=ET(KNT-1,2)
      CALL INTGRL(DX,DY,ET1,ET2,PPI,ALPHA)
      DXX=DX*DSTNRM
      DYY=DY*DSTNRM
      ETT1=ET1*PSINRM/DSTNRM
      ETT2=ET2*PSINRM/DSTNRM
      WRITE(6,1050) KNT,DXX,DYY,ETT1,ETT2,ALPHA,PPI
 1050 FORMAT(2X,I3,1X,1P6E10.2)
      TST=(PPI-PPI0)/PPI
      IF(TST.LT.1.E-4) GO TO(320,350),KEY
      GO TO 370
C
C     TEST FOR CONVERGENCE OF PSINRM. IF NOT, FIND NEW SOLUTION OF
C     POISSON'S EQ & REEVALUATE THE VALUE OF PSINRM REQUIRED FOR BREAK-
C     DOWN. IF SO, RENORMALIZE RAD,XWIDTH,YDEPTH,& QSS FOR THE
C     PURPOSE OF PROVIDING CORRECT OUTPUT DATA ON DIMENSIONS.
C
  380 TST=ABS((PSINRM-SINRM0)/SINRM0)
      WRITE(6,1060) KSEC
 1060 FORMAT(2X,//,2X,I3,1X,'ITERATIONS WERE REQUIRED WITH THE METHOD',
     &1X,'OF SECANTS IN AVLNCH',//)
      IF(TST.LE.VLTDLT) GO TO 390
      GO TO 100
  390 RAD=RDSAVE/(DSTNRM*1.E4)
      XWIDTH=XWSAVE/(DSTNRM*1.E4)
      YDEPTH=YDSAVE/(DSTNRM*1.E4)
      QSS=QSSAVE*DSTNRM/(11.7*8.854E-14*PSINRM)
      CALL GRID
      CALL STONE1(IMAX,JMAX)
      CALL BORDER
      CALL NITIAL
      CALL GRELAX
      CALL EFIELD
      RETURN
      END
C
C
```

```
      SUBROUTINE VGUESS(UNIT)
C
C     SUPPLIES INITIAL ESTIMATE OF THE BREAKDOWN VOLTAGE USING AN
C     EMPIRICAL FIT TO THE KENNEDY-O'BRIEN CURVES.
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      RR=RAD
      IF(RR.EQ.0) RR=0.1
      AA=1./(18.+150.*RR)
      UNIT=1./(AA+3.75E-12*DNSNRM**0.6)
      AA=2.303E4/(DNSNRM**.3695)
      BB=7.891E3/(DNSNRM**.2968)
      TT=TMPTR/300.-1.
      UNIT=UNIT*(AA*TT*TT+BB*TT+1.)
      RETURN
      END
C
C
```

```
      SUBROUTINE EQULIB
C
C     USING UNNORMALIZED VARIABLES, CALCULATES THE TEMPERATURE-
C     DEPENDENT BAND GAP, THE IONIZED ACCEPTOR DENSITY,AND THE BUILT-IN
C     POTENTIAL BARRIER, ACCOUNTING FOR THE TEMPERATURE DEPENDENCE OF
C     THE INTRINSIC DENSITY AND OF THE FERMI LEVEL.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
      COMMON/SWTCH/ NDOPE
C
      T=TMPTR/3.E2
      BLTZMN=0.02585*T
      EGAP=1.165-7.242E-3*T-3.664E-2*T*T
      EIV=EGAP/2.E0-1.306E-2*T
      X=EIV
      KEY=1
      GO TO 170
C
C     ADJUST X1&X2 UNTIL Y1&Y2 HAVE OPPOSITE SIGN
C
  100 X1=X
      Y1=Y
      DX=-EIV/2.E1
      IF(Y.LT.0.) DX=-DX
      X=X1+DX
      KEY=2
      GO TO 170
  110 X2=X
      Y2=Y
      IF(Y1*Y2.LE.0.) GO TO 140
  130 X1=X2
      Y1=Y2
      X=X2+DX
      GO TO 170
  140 KEY=3
C
C     METHOD OF SECANTS
C
  150 IF(ABS(Y).LT.1.E-3) GO TO 180
      DEN=Y2-Y1
      X=(X1*Y2-X2*Y1)/DEN
      GO TO 170
  160 X1=X2
      Y1=Y2
      X2=X
```

```
      Y2=Y
      GO TO 150
C
C     SUBROUTINE FOR EVALUATING Y
C
  170 EIF=X
      EFV=EIV-EIF
      EAF=4.38E-2-3.037E-8*ACCPTR**(1.0/3.0)-EFV
      BKGRND=ACCPTR/(1.0+(4.0+2.0/EXP(4.4E-2/BLTZMN))*EXP(EAF/BLTZMN))
      IF(NDOPE.GT.0) BKGRND=ACCPTR/(1.+2.*EXP(EAF/BLTZMN))
      TRNSIC=3.925E19*T*SQRT(T)*EXP(-EGAP/(2.*BLTZMN))
      BULKP=TRNSIC*EXP(EIF/BLTZMN)
      Y=(BULKP-BKGRND)/ACCPTR
      GO TO(100,110,160),KEY
C
C     CALCULATE THE BUILTIN POTENTIAL, THE ELECTRON DENSITY AT THE
C     METAL-SILICON INTERFACE, AND THE'BULK HOLE DENSITY.
C
  180 VBLTIN=BRRIER-EFV
      ONSNRM=BKGRND
      WRITE(6,1000)TMPTR,EGAP,EIV,EIF,EFV,TRNSIC,BKGRND,VBLTIN
 1000 FORMAT(2X,//,3X,'TMPTR',5X,'EGAP',7X,'EIV',7X,'EIF',7X,'EFV',
     &6X,'TRNSIC',4X,'BKGRND',4X,'VBLTIN',//,1X,1PSE10.2,//)
      RETURN
      END
C
C
```

```
      SUBROUTINE INDEX(I,J,DLTX,DLTY,KEXIT)
C
C     TRACKS WHICH 'BOX' OF THE LATTICE IS OCCUPIED BY THE MOVING
C     CHARGE & SUPPLIES ITS INDICES (I,J) AND CORRECT VALUES FOR
C     DLTX & DLTY WITH RESPECT TO THAT POINT. ALSO FLAGS AN EXIT OF
C     THE CHARGE FROM THE REGION.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      IF(DLTX.LT.0.) GO TO 100
      IF(DLTX.GT.XDLT(I+1)) GO TO 110
      GO TO 120
  100 IF(I.EQ.1) GO TO 150
      DLTX=XDLT(I)+DLTX
      I=I-1
      IF(DLTX.LT.0.) GO TO 100
      GO TO 120
  110 IF(I.EQ.(IMAX-1)) GO TO 150
      DLTX=DLTX-XDLT(I+1)
      I=I+1
      IF(DLTX.GT.XDLT(I+1)) GO TO 110
  120 IF(DLTY.LT.0.) GO TO 130
      IF(DLTY.GT.YDLT(J)) GO TO 140
      RETURN
  130 IF(J.EQ.JMAX) GO TO 150
      DLTY=YDLT(J+1)+DLTY
      J=J+1
      IF(DLTY.LT.0.) GO TO 130
      RETURN
  140 IF(J.EQ.2) GO TO 150
      DLTY=DLTY-YDLT(J)
      J=J-1
      IF(DLTY.GT.YDLT(J)) GO TO 140
      RETURN
  150 KEXIT=1
      RETURN
      END
C
C
```

```
      SUBROUTINE EFORCE(I,J,X,Y,DLTX,DLTY,EX,EY,ET)
C
C     APPLIES INTERPOLATION FOR THE X & Y COMPONENTS OF THE ELECTRIC
C     FIELD AT LOCATION (X,Y).
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
  100 EN=ESUBX(I,J)+DLTX*(ESUBX(I+1,J)-ESUBX(I,J))/XDLT(I+1)
      ES=ESUBX(I,J-1)+DLTX*(ESUBX(I+1,J-1)-ESUBX(I,J-1))/XDLT(I+1)
      EX=EN+DLTY*(ES-EN)/YDLT(J)
      EN=ESUBY(I,J)+DLTX*(ESUBY(I+1,J)-ESUBY(I,J))/XDLT(I+1)
      ES=ESUBY(I,J-1)+DLTX*(ESUBY(I+1,J-1)-ESUBY(I,J-1))/XDLT(I+1)
      EY=EN+DLTY*(ES-EN)/YDLT(J)
      ET=SQRT(EX*EX+EY*EY)
      RETURN
      END
C
C
```

```
      SUBROUTINE TRAJEC(KNT,I,J,X,Y,DLTX,DLTY,ET,KEXIT)
C
C     USES ADAMS-BASHFORTH-MOULTON PREDICTOR-CORRECTOR EQ'S TO TRACE
C     HOLE TRAJECTORIES, FOLLOWING A FOURTH-ORDER RUNGE-KUTTA STARTING
C     PROCEDURE.
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      DIMENSION DX(7),DY(7)
C
      IF(KNT.GT.4) GO TO 100
      IF(KNT.GT.1) GO TO 90
      DLTA=0.005
      TDELT=DLTA/ET
      DO 80 K=1,7
      DX(K)=0.
   80 DY(K)=0.
C
C     RUNGE-KUTTA PROCEDURE
C
   90 CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      CALL EFORCE(I,J,X,Y,DLTX,DLTY,EX,EY,ET)
      DX1=EX*TDELT
      DY1=EY*TDELT
      XX=X+DX1/2.E0
      YY=Y+DY1/2.E0
      DLTX=XX-XPOS(I)
      DLTY=YY-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      CALL EFORCE(I,J,XX,YY,DLTX,DLTY,EX,EY,ET)
      DX2=EX*TDELT
      DY2=EY*TDELT
      XX=X+DX2/2.E0
      YY=Y+DY2/2.E0
      DLTX=XX-XPOS(I)
      DLTY=YY-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      CALL EFORCE(I,J,XX,YY,DLTX,DLTY,EX,EY,ET)
      DX3=EX*TDELT
      DY3=EY*TDELT
      XX=X+DX3
      YY=Y+DY3
      DLTX=XX-XPOS(I)
      DLTY=YY-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
```

```
      CALL EFORCE(I,J,XX,YY,DLTX,DLTY,EX,EY,ET)
      DX4=EX*TDELT
      DY4=EY*TDELT
      DX(KNT+3)=(DX1+2.E0*(DX2+DX3)+DX4)/6.E0
      DY(KNT+3)=(DY1+2.E0*(DY2+DY3)+DY4)/6.E0
      X=X+DX(KNT+3)
      Y=Y+DY(KNT+3)
      DLTX=X-XPOS(I)
      DLTY=Y-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      CALL EFORCE(I,J,X,Y,DLTX,DLTY,EX,EY,ET)
      DX(KNT+3)=EX*TDELT
      DY(KNT+3)=EY*TDELT
      KOBLE=3
      RETURN
C
C     PREDICTOR-CORRECTOR, USING THE PRESENT INTERVAL
C
  100 XPROCT=(55.E0*DX(7)-59.E0*DX(6)+37.E0*DX(5)-9.E0*DX(4))/24.E0
      YPROCT=(55.E0*DY(7)-59.E0*DY(6)+37.E0*DY(5)-9.E0*DY(4))/24.E0
      XX=X+XPROCT
      YY=Y+YPROCT
      DLTX=XX-XPOS(I)
      DLTY=YY-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      IF(KEXIT.GT.0) RETURN
      CALL EFORCE(I,J,XX,YY,DLTX,DLTY,EX,EY,ET)
      DX1=EX*TDELT
      DY1=EY*TDELT
      XCRRCT=(9.E0*DX1+19.E0*DX(7)-5.E0*DX(6)+DX(5))/24.E0
      YCRRCT=(9.E0*DY1+19.E0*DY(7)-5.E0*DY(6)+DY(5))/24.E0
      DX2=ABS(XPROCT-XCRRCT)
      DY2=ABS(YPROCT-YCRRCT)
      DLT=AMAX1(DX2,DY2)
C
C     HALVE THE INTERVAL IF DLT EXCEEDS DLTA
C
      IF(DLT.LT.DLTA) GO TO 110
C
C     PARABOLIC INTERPOLATION COEFFICIENTS FOR THE HALF INTERVAL VALUES
C
      KOBLE=3
      TDELT=TDELT/2.E0
      AA=(DX(7)-2.E0*DX(6)+DX(5))/(8.E0*TDELT*TDELT)
      BB=(DX(7)-DX(5))/(4.E0*TDELT)
      AAA=(DX(6)-2.E0*DX(5)+DX(4))/(8.E0*TDELT*TDELT)
      BBB=(DX(6)-DX(4))/(4.E0*TDELT)
      DX(1)=DX(4)/2.E0
      DX(3)=DX(5)/2.E0
      DX(5)=DX(6)/2.E0
      DX(2)=(TDELT*(AAA*TDELT-BBB)+DX(3))/2.E0
      DX(4)=(TDELT*(AA*TDELT-BB)+DX(5))/2.E0
      DX(6)=(TDELT*(AA*TDELT+BB)+DX(5))/2.E0
      DX(7)=DX(7)/2.E0
      AA=(DY(7)-2.E0*DY(6)+DY(5))/(8.E0*TDELT*TDELT)
      BB=(DY(7)-DY(5))/(4.E0*TDELT)
```

```
      AAA=(DY(6)-2.E0*DY(5)+DY(4))/(8.E0*TDELT*TDELT)
      BBB=(DY(6)-DY(4))/(4.E0*TDELT)
      DY(1)=DY(4)/2.E0
      DY(3)=DY(5)/2.E0
      DY(5)=DY(6)/2.E0
      DY(2)=(TDELT*(AAA*TDELT-BBB)+DY(3))/2.E0
      DY(4)=(TDELT*(AA*TDELT-BB)+DY(5))/2.E0
      DY(6)=(TDELT*(AA*TDELT+BB)+DY(5))/2.E0
      DY(7)=DY(7)/2.E0
      GO TO 100
C
C     TENTATIVELY DOUBLE THE INTERVAL,BUT ONLY IF 3 STEPS HAVE BEEN
C     TAKEN SINCE THE LAST DOUBLING OR HALVING, AND ONLY IF DLT
C     IS LESS THAN ONE HALF OF DLTA
C
  110 IF(KOBLE.GT.0) GO TO 120
      DLT=DLT*2.0
      IF(DLT.GT.DLTA) GO TO 120
      XPRDT=(55.E0*DX(7)-59.E0*DX(5)+37.E0*DX(3)-9.E0*DX(1))/12.E0
      YPRDT=(55.E0*DY(7)-59.E0*DY(5)+37.E0*DY(3)-9.E0*DY(1))/12.E0
      XX=X+XPRDT
      YY=Y+YPRDT
      DLTX=XX-XPOS(I)
      DLTY=YY-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      IF(KEXIT.GT.0) GO TO 120
      CALL EFORCE(I,J,XX,YY,DLTX,DLTY,EX,EY,ET)
      DX1=EX*TDELT
      DY1=EY*TDELT
      XCRRT=(9.E0*DX1+19.E0*DX(7)-5.E0*DX(5)+DX(3))/12.E0
      YCRRT=(9.E0*DY1+19.E0*DY(7)-5.E0*DY(5)+DY(3))/12.E0
      DX2=ABS(XPRDT-XCRRT)
      DY2=ABS(YPRDT-YCRRT)
      DLT=AMAX1(DX2,DY2)
      IF(DLT.GT.DLTA) GO TO 120
C
C     DOUBLE INTERVAL IS OK, SO DO IT
C
      DX(7)=DX(7)*2.E0
      DX(6)=DX(5)*2.E0
      DX(5)=DX(3)*2.E0
      DX(4)=DX(1)*2.E0
      DY(7)=DY(7)*2.E0
      DY(6)=DY(5)*2.E0
      DY(5)=DY(3)*2.E0
      DY(4)=DY(1)*2.E0
      X=X+(19.E0*XPRDT+251.E0*XCRRT)/270.E0
      Y=Y+(19.E0*YPRDT+251.E0*YCRRT)/270.E0
      TDELT=2.E0*TDELT
      KOBLE=3
      GO TO 130
C
C     NO DOUBLING OCCURRED. COMPUTE NEW X & Y. SHIFT INCREMENTS LEFT
C     & UPDATE THE MOST RECENT INCREMENT.
C
  120 X=X+(19.E0*XPRDCT+251.E0*XCRRCT)/270.E0
```

```
      Y=Y+(19.E0*YPRDCT+251.E0*YCRRCT)/270.E0
      KOBLE=KOBLE-1
  130 DO 140 K=1,6
      DX(K)=DX(K+1)
  140 DY(K)=DY(K+1)
      DLTX=X-XPOS(I)
      DLTY=Y-YPOS(J)
      CALL INDEX(I,J,DLTX,DLTY,KEXIT)
      IF(KEXIT.GT.0) RETURN
      CALL EFORCE(I,J,X,Y,DLTX,DLTY,EX,EY,ET)
      ETT=ET*PSINRM/DSTNRM
      IF(ETT.LT.2.E4) KEXIT=1
      DX(7)=EX*TDELT
      DY(7)=EY*TDELT
      RETURN
      END
C
C
```

```
      SUBROUTINE INTGRL(DX,DY,E1,E5,PI,ALPHA)
C
C     USES THE KENNEDY IONIZATION COEFFICIENT (AT 300 DEG K), SCALED TO
C     THE TEMPERATURE T USING CROWELL & SZE'S EMPIRICAL FIT TO BARAFF'S
C     CURVES, TO EVALUATE THE IONIZATION INTEGRAL BY GAUSSIAN QUADRATURE
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31).
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/ONSNRM,PSINRM,DSTNRM,BLTZMN
C
      DS=SQRT(DX*DX+DY*DY)*DSTNRM
      DZ=DS*A1/2.
      AA=(E5-E1)/DS
      BB=(E5+E1)/2.
      E2=(-AA*DZ+BB)*PSINRM/DSTNRM
      E3=BB*PSINRM/DSTNRM
      E4=(AA*DZ+BB)*PSINRM/DSTNRM
      AA=1.47E6
      BB=1.875E6
      ET=E2
      ALPHA=AA/EXP(BB/E2)
      KEY=1
      GO TO 130
  100 PI=PI+ALPHA*H1*DS/2.
      ET=E3
      ALPHA=AA/EXP(BB/E3)
      KEY=2
      GO TO 130
  110 PI=PI+ALPHA*H2*DS/2.
      ET=E4
      ALPHA=AA/EXP(BB/E4)
      KEY=3
      GO TO 130
  120 PI=PI+ALPHA*H1*DS/2.
      RETURN
  130 IF(TMPTR.EQ.300.) GO TO(100,110,120),KEY
      BLTZ=.02585
      CALL CRWSZE(ET,BLTZ,ALPH)
      RATIO=ALPHA/ALPH
      BLTZ=BLTZMN
      CALL CRWSZE(ET,BLTZ,ALPH)
      ALPHA=RATIO*ALPH
      GO TO(100,110,120),KEY
      ENTRY INTGR1
      A1=.774596669241483
      H1=.555555555555556
      H2=.888888888888889
      RETURN
      END
```

```
      SUBROUTINE CRWSZE(ET,BLTZ,ALPH)
C
C     CALCULATES THE TOWNSEND COEFFICIENT FOR ELECTRONS USING CROWELL &
C     SZE'S EMPIRICAL FIT FOR BARAFF'S CURVES. (USES UNNORMALIZED
C     VARIABLES.)
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIOLT
C
      EXPNT(R2,R1,X2,X1)=(11.5*R2-1.17*R1+3.9E-4)*X2+
     &(46.*R2-11.9*R1+1.75E-2)*X1-(757.*R2-75.5*R1+1.92)
      TANH(X)=(EXP(X)-EXP(-X))/(EXP(X)+EXP(-X))
      XX=3.15E-2/BLTZ
      EP=6.3E-2*TANH(XX)
      ELMBDA=76.0E-8*EP/6.3E-2
      EI=1.5*EGAP
      R1=EP/EI
      R2=R1*R1
      X1=EI/(ET*ELMBDA)
      X2=X1*X1
      XN=EXPNT(R2,R1,X2,X1)
      IF(XN.LT.-80.) XN=-80.
      ALPH=EXP(XN)/ELMBDA
      RETURN
      END
C
C
```

```
      SUBROUTINE OUTPUT
C
C
C
      COMMON/ARRAYS/PSI(31,31),EDENS(31,31),HDENS(31,31),B(31,31),
     &D(31,31),E(31,31),F(31,31),H(31,31),Q(31,31),QAREA(31,31),
     &DELTA(31,31),ESUBX(31,31),ESUBY(31,31),IDENT(31,31)
      COMMON/COEFF/HSTH(31,31),HWEST(31,31),HCNTR(31,31),HEAST(31,31),
     &HNRTH(31,31),XDLT(32),YDLT(32),XPOS(31),YPOS(31)
      COMMON/REAL/XWIDTH,YDEPTH,BRRIER,EGAP,TMPTR,ACCPTR,VBLTIN,VOLT,
     &TRNSIC,QSS,X0,Y0,RAD,XDELTA,YDELTA
      COMMON/INTGR/IMAX,JMAX,IXWDTH,JYDPTH
      COMMON/CNTRL/ITMAX,ITRMAX,CONVRG,SIDLT
      COMMON/NRMLZE/DNSNRM,PSINRM,DSTNRM,BLTZMN
C
      DIMENSION SCRTCH(31),X(31),Y(31)
C
      DO 80 I=1,IMAX
   80 X(I)=XPOS(I)*DSTNRM*1.0E4
      DO 90 J=1,JMAX
   90 Y(J)=YPOS(J)*DSTNRM*1.0E4
      WRITE(6,1000)
 1000 FORMAT(1H1,60X,/,32X,'X(I),I=1,IMAX',//)
      WRITE(6,1010)(X(I),I=1,IMAX)
 1010 FORMAT(32X,1P10E10.2)
      WRITE(6,1020)
 1020 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'PSI(I,J),I=1,IMAX',//)
      DO 110 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 100 I=1,IMAX
  100 SCRTCH(I)=PSI(I,J)*PSINRM
      WRITE(6,1030) J,Y(J),(SCRTCH(I),I=1,10)
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
 1030 FORMAT(1X,' ',/,I7,5X,1PE10.2,10X,1P10E10.2)
  110 CONTINUE
 1040 FORMAT(32X,1P10E10.2)
      WRITE(6,1000)
      WRITE(6,1010)(X(I),I=1,IMAX)
      WRITE(6,1060)
 1060 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'HDENS(I,J),I=1,IMAX',//)
      DO 150 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 140 I=1,IMAX
  140 SCRTCH(I)=HDENS(I,J)*DNSNRM
      WRITE(6,1030) J,Y(J),(SCRTCH(I),I=1,10)
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
  150 CONTINUE
      WRITE(6,1000)
      WRITE(6,1010)(X(I),I=1,IMAX)
      WRITE(6,1100)
 1100 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'EDENS(I,J),I=1,IMAX',//)
      DO 154 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 152 I=1,IMAX
  152 SCRTCH(I)=EDENS(I,J)*DNSNRM
      WRITE(6,1030)J,Y(J),(SCRTCH(I),I=1,10)
```

```
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
  154 CONTINUE
      WRITE(6,1000)
      WRITE(6,1010)(X(I),I=1,IMAX)
      WRITE(6,1070)
 1070 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'ESUBX(I,J),I=1,IMAX',//)
      DO 170 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 160 I=1,IMAX
  160 SCRTCH(I)=ESUBX(I,J)*PSINRM/DSTNRM
      WRITE(6,1030) J,Y(J),(SCRTCH(I),I=1,10)
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
  170 CONTINUE
      WRITE(6,1000)
      WRITE(6,1010)(X(I),I=1,IMAX)
      WRITE(6,1080)
 1080 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'ESUBY(I,J),I=1,IMAX',//)
      DO 190 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 180 I=1,IMAX
  180 SCRTCH(I)=ESUBY(I,J)*PSINRM/DSTNRM
      WRITE(6,1030) J,Y(J),(SCRTCH(I),I=1,10)
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
  190 CONTINUE
      WRITE(6,1000)
      WRITE(6,1010)(X(I),I=1,IMAX)
      WRITE(6,1090)
 1090 FORMAT(1H0,5X,'J',9X,'Y(J)',15X,'ETOTAL(I,J),I=1,IMAX',//)
      DO 210 JJ=1,JMAX
      J=JMAX+1-JJ
      DO 200 I=1,IMAX
  200 SCRTCH(I)=SQRT(ESUBX(I,J)*ESUBX(I,J)+ESUBY(I,J)*ESUBY(I,J))*
     &PSINRM/DSTNRM
      WRITE(6,1030) J,Y(J),(SCRTCH(I),I=1,10)
      WRITE(6,1040)(SCRTCH(I),I=11,IMAX)
  210 CONTINUE
      RETURN
      END
```

MISSION
of
Rome Air Development Center

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control Communications and Intelligence (C^3I) activities. Technical and engineering support within areas of technical competence is provided to ESD Program Offices (POs) and other ESD elements. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.

Printed by
United States Air Force
Hanscom AFB, Mass. 01731